1974-12

Validation of the nonlinear six degree freedom mathematical model of the XR-3 captured air bubble surface effect ship in calm water.

Forbes, George Thomas.

VALIDATION OF THE NONLINEAR SIX DEGREE OF FREEDOM MATHEMATICAL MODEL OF THE XR-3 CAPTURED AIR BUBBLE SURFACE EFFECT SHIP IN CALM WATER

George Thomas Forbes

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

VALIDATION OF THE NONLINEAR SIX DEGREE OF FREEDOM MATHEMATICAL MODEL OF THE XR-3 CAPTURED AIR BUBBLE SURFACE EFFECT SHIP IN CALM WATER

by

George Thomas Forbes

December 1974

Thesis Advisor: A. Gerba, Jr.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) VALIDATION OF THE NONLINEAR SIX DEGREE OF FREEDOM MATHEMATICAL MODEL OF THE XR-3 CAPTURED AIR BUBBLE SURFACE EFFECT SHIP IN CALM WATER		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; December 1974
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) George Thomas Forbes		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE December 1974
		13. NUMBER OF PAGES 132
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Naval Postgraduate School Monterey, California 93940		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The non-linear, six-degree-of-freedom, mathematical model of the XR-3 captured air bubble surface effect ship is validated by comparison of measured testcraft performance with computed variables in the digital computer simulation. Data processing methods implemented on hybrid and digital computers are presented. Improvements are made on the structure of the mathematical model of the XR-3 by program subroutine modifications and justification

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

is given for the changes made. Close agreement is obtained between measured and computer simulated variables over a variety of operating conditions in calm water.

Validation of the Nonlinear Six Degree of Freedom
Mathematical Model of the XR-3 Captured Air Bubble
Surface Effect Ship in Calm Water

by

George Thomas Forbes
Lieutenant Commander, United States Navy
B.S., United States Naval Academy, 1965

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 1974

ABSTRACT

The non-linear, six-degree-of-freedom, mathematical model of the XR-3 captured air bubble surface effect ship is validated by comparison of measured testcraft performance with computed variables in the digital computer simulation. Data processing methods implemented on hybrid and digital computers are presented. Improvements are made on the structure of the mathematical model of the XR-3 by program subroutine modifications and justification is given for the changes made. Close agreement is obtained between measured and computer simulated variables over a variety of operating conditions in calm water.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF DRAWINGS

TABLE OF SYMBOLS AND ABBREVIATIONS

ac	alternating current
A-to-D	analog to digital
CAB	Captured Air Bubble
CG	Center of Gravity
℄	Center Line
db	decibels
dc	direct current
deg	degrees
D-to-A	digital to analog
FXPWAV	drag force in direction opposite to craft motion due to drag between pressurized air and water surface
g	gravitational acceleration
Hz	Hertz
kts	knots
lb	pounds
NPS	U.S. Naval Postgraduate School
NSRDC	Naval Ship Research and Development Center
RMS	Root Mean Square
psf	pounds per square foot
sq ft	square feet
ρ	density
δ	rudder angle
θ	pitch angle
ϕ	roll angle

ψ	yaw angle
plenum	cavity beneath the wet deck of the XR-3, formed by the sidewalls and the bow and stern seals, into which pressurized air is blown
hump speed	speed at which testcraft overcomes large low velocity drag and begins to operate on the cushion of cpatured pressurized air

ACKNOWLEDGEMENT

The author wishes to express his sincere appreciation to Associate Professor Alex Gerba of the U.S. Naval Postgraduate School for the guidance, assistance, and encouragement which he provided during the pursuit of this study.

The author wishes to thank Mr. Robert Limes of the U.S. Naval Postgraduate School Computer Laboratory for his valuable assistance in data processing. Associate Professor D.M. Layton and his technical assistant, Mr. Michael O'Dell were of tremendous assistance in the collection of data.

I. INTRODUCTION

A. BACKGROUND

The Captured Air Bubble surface effect ship presents a new concept in shipping for the surface force of the U.S. Navy. Within a few years these vessels, supported partially by a cushion of air and partially by the more conventional hull buoyancy forces, will be joining the Fleet. In preparation for that event it is of extreme importance for detailed studies to be undertaken so that the capabilities and limitations of such vessels may be fully understood.

The term surface effect ship applies to a broad spectrum of different types of vessels. Among them are the Air Cushion Vehicle (ACV), the Ground Effects Machine (GEM), the Semi-Submerged Ship (SSS), the Captured Air Bubble (CAB), and others. This study is concerned with the CAB surface effect ship.

The Captured Air Bubble craft consists of two rigid sidewalls which resemble catamaran hulls and flexible seals between the water surface and hull structure across the bow and stern of the craft. The enclosed volume is a plenum chamber into which air is blown by a lift fan system. The pressure in the plenum chamber acting over the area of the plenum provides a lift force sufficient to support a large proportion of the total craft weight. The remainder

of the supporting force is made up by conventional hydrodynamic forces acting on the sidewalls, plus some lift forces due to the bow and stern seals and aerodynamic lift at higher operating speeds.

In the "on bubble" mode of operation, at the higher operating speeds, approximately two-thirds of the total craft weight is supported by the air cushion, and therefore relatively shallow sidewall immersion is required to support the remaining weight. The shallow draft produces a lower hydrodynamic drag and thus higher speed for a given power than could be obtained for a displacement hull.

The XR-3 is a CAB surface effect ship of approximately three tons displacement. The XR-3 was built by the Naval Ship Research and Development Center (NSRDC) in 1966. Following periods of testing by NSRDC, Annapolis Division NSRDC, and Aerojet-General Corporation, the XR-3 was delivered to the Naval Postgraduate School (NPS), Monterey, California in March 1970 for the investigation of basic and advanced surface effect ship technology.

In addition to the XR-3 there exist other CAB test craft. Among these are the Aerojet-General 100-A and the Bell Aerospace Systems 100-B, both of approximately 100 tons displacement. These two vessels were constructed as prototypes of a 2,000 ton ship. To date speeds in the 80-knot range have been attained by these vessels.

Under a U.S. Government contract, Oceanics Incorporated developed a digital computer simulation of the

CAB ship loads and motions for the Surface Effect Ship Project Office (SESPO) in Washington D.C. This loads and motions program with the mathematical model of the Bell Aerospace Systems 100-B with six-degrees-of-freedom has been in use at the W.R. Church Computer Center at NPS since October 1972.

In December 1973, LCDR. D.G. Leo, USN and LT. R. Boncal, USN produced a mathematical model with six-degrees-of-freedom of the XR-3 test craft derived from the above-mentioned simulation model of the 100-B (Ref. 1). The modular construction of the 100-B simulation program facilitated the re-modeling of the program where necessary to fit the design differences between the 100-B and the XR-3.

B. OBJECTIVES

It is the objective of this thesis to validate the non-linear, six-degrees-of-freedom, mathematical model of the XR-3 CAB test craft in a comprehensive manner by direct comparison between instrument-measured variables on the XR-3 and computed variables produced by the simulation programs. In the process of carrying out this objective, subroutine modifications were made to the simulation program to improve on the structure of the simulation model and justification is given in the sections that follow for the changes that were made.

The data taken on the XR-3 measured in real-time, analog fashion using FM modulation was filtered, converted to

digital form, and processed in such a manner that it could be interfaced directly with the simulation program and a time history of true differences for each variable could be produced at each operating condition.

The present study is limited to operation of the test craft in calm water conditions at the test site located at Lake San Antonio Reservoir, California. The range of operation of the test craft includes:

1. Straight runs at various speeds with various locations of the center of gravity. The center of gravity was shifted by placement of two 125 pound water tanks at different locations on the deck of the test craft.
2. Straight runs at various speeds in an unladen condition.
3. Runs made at various speeds with various rudder angles applied to validate turning performance.

II. COLLECTION OF DATA

A. INSTRUMENTATION ON THE XR-3

In order to support the continuing studies being undertaken at the Naval Postgraduate School on Captured Air Bubble vessels, the XR-3 has been instrumented with devices capable of measuring and recording time histories of many different variables for later, post-run analysis (Ref.2). The general configuration of the XR-3 is shown in Figure 1. Table I shows a listing of the measurable parameters, the range of measurement, and the accuracy with which the measurements may be made. The locations of the measuring instruments are shown in Table II. The measured data are recorded on a Pemco Model 120-B magnetic tape recorder which features 14 tracks for recording data and an edge track for voice annotation during measurement runs. Each of the 14 tracks on the tape recorder is capable of recording $\pm$ 1 volt RMS with a tolerance of $\pm$ 0.5%.

B. POWER SUPPLIES

The tape recorder is powered from a 110 volt 60 cycle auxiliary power unit, located on the aft weather deck as shown in Figure 1., through a Pemco a.c. to d.c. converter which provides a 26 volt d.c. power source. The three accelerometers for measuring vertical, lateral, and fore-and-aft accelerations are powered from a 12 volt storage battery through a d.c. to d.c. converter which provides a

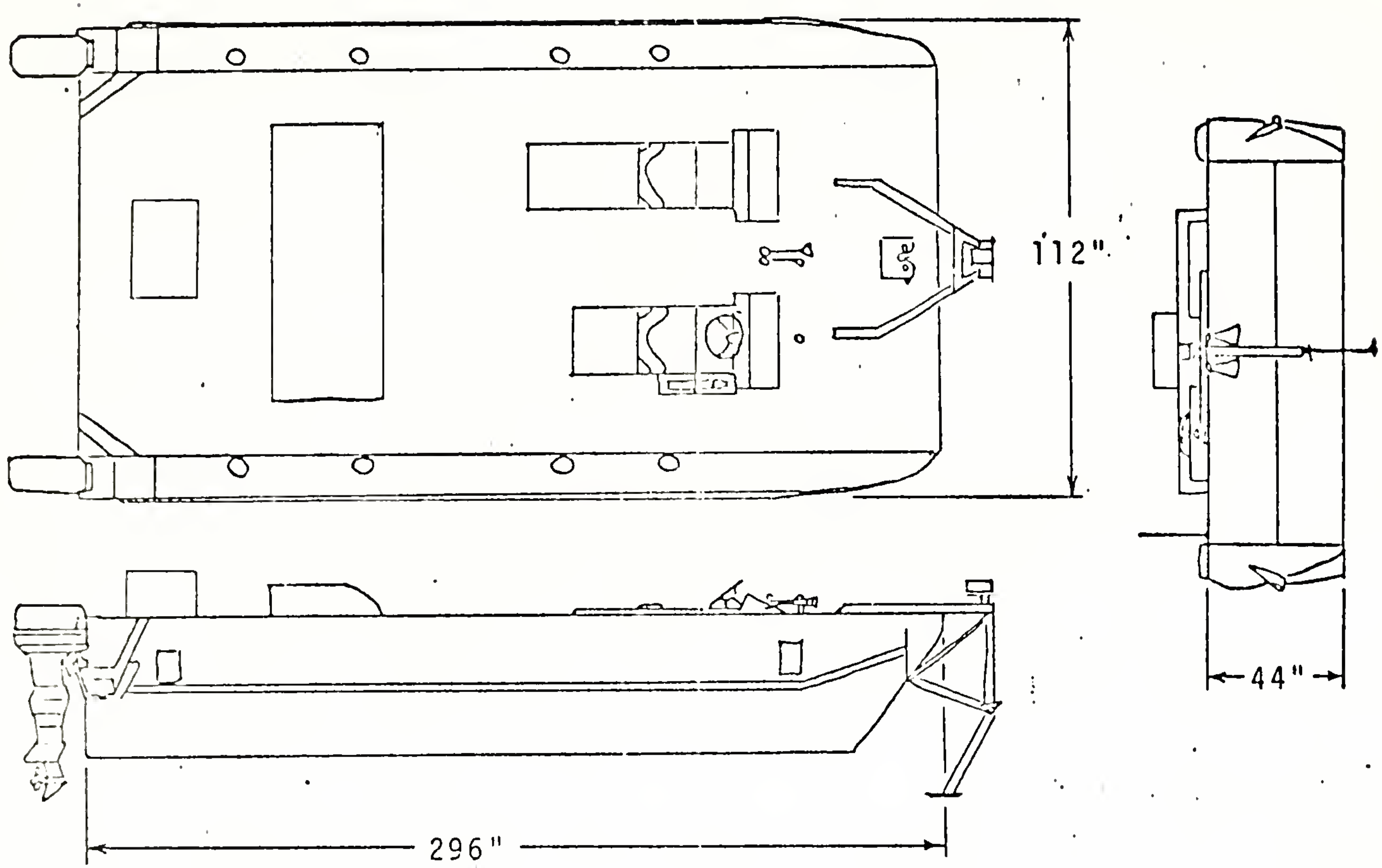


FIGURE 1. GENERAL CONFIGURATION OF XR-3

Table I. XR-3 Measurable Parameters

Parameter	Simulation Program Name	Range	Accuracy
Port Thrust	THSTP	0-500 lbs	± 5 lb
Starboard Thrust	THSTS	0-500 lbs	± 5 lb
Velocity	VEL	0-40 kts	± 1 kt
Height	None*	± 2 feet	± 0.1in
Bowseal Press.	PBARB	0-60 psf	± 0.5 psf
Plenum Press.	PBAR	0-60 psf	± 0.5 psf
Sternseal Press.	PBARS	0-60 psf	± 0.5 psf
Pitch Angle	THETAR	± 15 degs	± 0.5 deg
Roll Angle	DPHI	± 15 degs	± 0.5 deg
Yaw Angle	DPSI	± 180 degs	± 0.5 deg
Pitch Rate	PDEG	±30 degs/sec	±0.5 deg/sec
Roll Rate	QDEG	±30 degs/sec	±0.5 deg/sec
Yaw Rate	RDEG	±30 degs/sec	±0.5 deg/sec
Rudder Position	DELRS	± 45 degs	not available
Longitudinal Acceleration	None	± 0.2g units	not available
Lateral Acceleration	ACCLAT	± 0.2g units	not available
Vertical Acceleration	ACCEL(3)	- 0.8 to +1.2g	not available

*Height is used to compute Draft as explained in text.

Table II. Location of Sensors

Sensor	Distance Forward of Transom	Distance from Center Line
Pressure Transducers		
Bow Seal	18'4"	℄
Plenum	12'4½"	℄
Stern Seal	1'6"	℄
Velocity	21'9" 4'9½" down from deck	℄
Height	24'1" 6" down from deck	℄
Gyros		
Pitch	12'6"	2'4½" Port
Roll	13'	2'4½" Port
Yaw	12'6"	2'4½" Port
Port Thrust	0	5'6" Port
Starboard Thrust	0	5'6" Starboard
Vertical Accelerometer	13'6½"	1'9½" Port
Lateral Accelerometer	13'6"	1'9" Port
Longitudinal Accelerometer	13'5½"	1'8½" Port

28 volt d.c. supply. The three gyros which measure pitch, roll, yaw, pitch rate, roll rate and yaw rate are powered by the 12 volt storage battery. The gyros also receive a 1 volt d.c. excitation signal from the 110 volt a.c. power supply fed through an a.c. to d.c. converter.

The height sensor mounted forward of the bow is powered directly from the 110 volt a.c. auxiliary power unit. The height measurement signal is converted to a $\pm$1.0 volt d.c. signal by a height sensor conditioner. The velocity probe, mounted on the same assembly as the height sensor, extends into the undisturbed water forward of the testcraft. The 28 volt d.c. power source for the velocity probe is the 12 volt battery through the d.c. to d.c. converter. The rudder position measurement is made by means of a load cell which receives a 12 volt d.c. supply from the battery.

The two load cells for measuring port and starboard thrust and the three pressure transducers for measuring bow seal, plenum, and stern seal pressures are all powered by a 28 volt d.c. supply from the 12 volt d.c. storage battery through the d.c. to d.c. converter. All five units also receive a 5 volt d.c. excitation signal from an amplifier-transducer package.

III. DATA PROCESSING

A. ANALOG TO DIGITAL CONVERSION

To provide measurements for validation of the mathematical model, data-gathering runs were made with the XR-3 testcraft on four different days at the Lake San Antonio, California test site. The data was recorded on magnetic tape using FM modulation, then processed into a suitable form for comparison with corresponding values computed in the XR-3 simulation program on the IBM 360/67 Digital Computer.

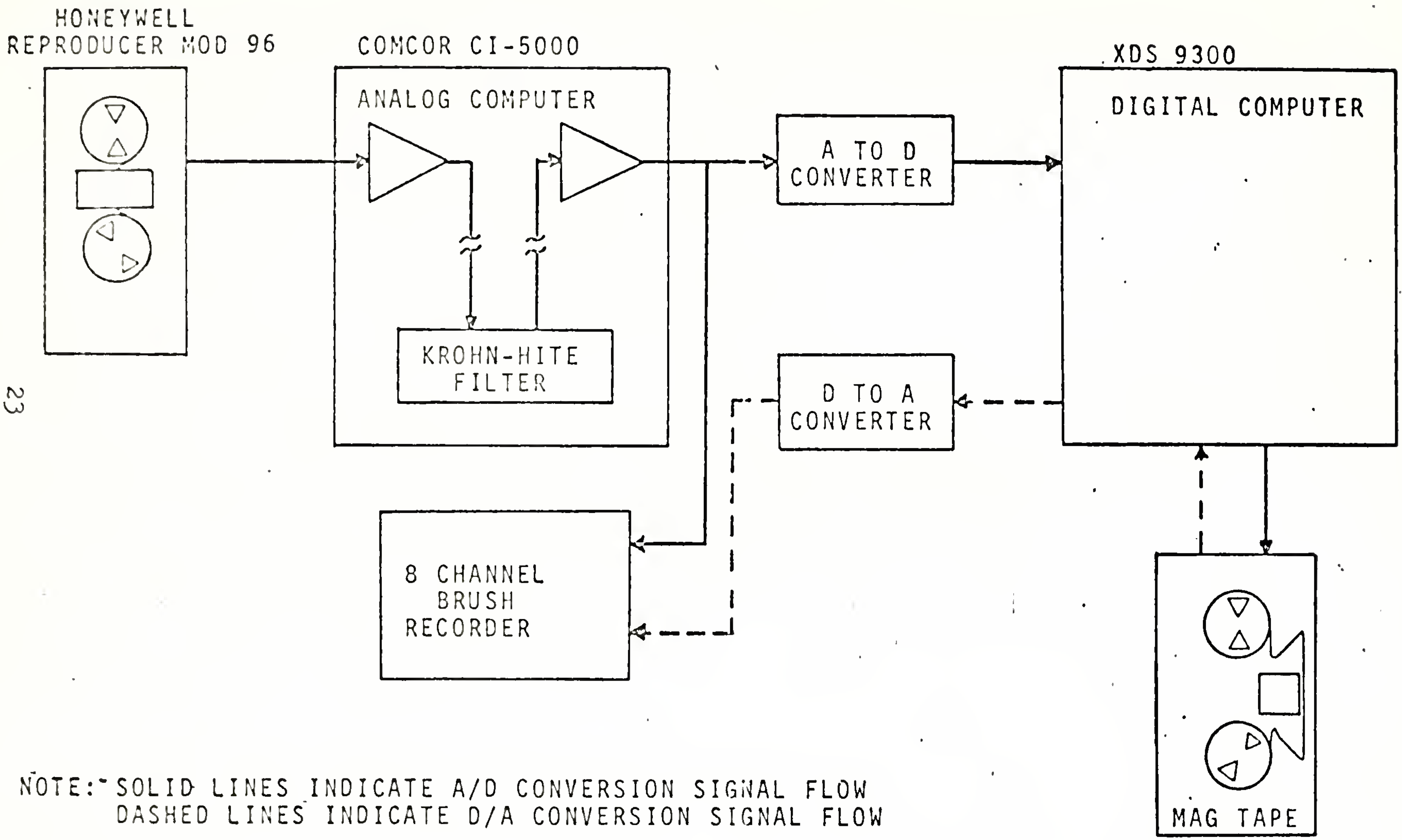
The data was converted from analog to digital form on an XDS-9300/CI-5000 hybrid computer at the NPS Computer Laboratory. From previous experience in data reduction by the XR-3 Research Group in the Aeronautical Engineering Department at NPS, it was found that filtering of the analog signal was required to remove unwanted high frequency noise from the recorded data. To provide this required filtering six Krohn-Hite digitally-tuned filters were assembled so that six tracks of recorded data could be processed simultaneously.

B. DESCRIPTION OF EQUIPMENT

The analog tape containing raw data from XR-3 test runs was mounted on a Honeywell Tape Recorder/Reproducer, Model 96 for playback. Connections were made through trunk lines to lead the output from the six selected tracks

to the COMCOR CI-5000 Analog Computer as shown in Figure 2. Each channel of recorded data was then passed through an amplifier with a gain of 5 before being led out of the CI-5000 to one of the filters. The filters were operated in the low pass-maximally flat mode which provided a four-pole Butterworth response with an attenuation of 24db per octave from the cut off frequency. Cut off frequencies were set at 10 Hertz for all channels except port and starboard thrust which were set at 1.0 Hertz. Each signal at the output of its respective filter was led back into the CI-5000 for another amplification by a gain factor of ten. A Brush 8-Channel Recorder was also patched into the system at this point so that each of the six selected data tracks could be continuously monitored. Figure 2 shows the signal flow patch through the hybrid computer system for analog-to-digital conversion.

For analog-to-digital conversion, digital-to-analog conversion, and other applications, the CI-5000 Analog Computer and the XDS-9300 Digital Computer are interconnected through trunks. For A-to-D conversion in this application a sampling frequency of 200 Hz was chosen. In steady-state calm-water runs with the XR-3 none of the measured parameters can be expected to vary at frequencies greater than 10 Hz. Therefore, selection of a 200 Hz sampling frequency more than satisfies the requirements of the Sampling Theorem.



NOTE: SOLID LINES INDICATE A/D CONVERSION SIGNAL FLOW
DASHED LINES INDICATE D/A CONVERSION SIGNAL FLOW

FIGURE 2. SIGNAL FLOW DURING ANALOG TO DIGITAL CONVERSION

The first step in the A-to-D process was to zero all possible bias errors in the system. Calibration steps are recorded on separate channel strips on the analog data tape which indicate zero and maximum signal readings for each sensor. Bias potentiometers at the input of the first stage amplifier on the CI-5000 were adjusted to properly zero all channels during the pre-test run calibration of the XR-3 sensors. Each signal was monitored on a digital voltmeter during the calibration procedure.

Execution of the FORTRAN program listed in Appendix A on the hybrid computer system enabled selective conversion from analog to digital form of specific time periods of analog data. Continuous monitoring of the voice edge track on the data tape provided a reference as to what the XR-3 conditions were at all times. After each file of data had been digitized and written on the digital magnetic tape, the tape was re-wound and that file most recently recorded was run through the digital-tc-analog conversion. The D-to-A signal was monitored on the Brush Recorder and compared with the analog signal to verify the validity of the conversion process. The finished product of the A-to-D conversion was a reel of 7-track digital tape consisting of many files of digital data. Each file on the tape contained a time history of six selected variables measured for a specific operating condition on the XR-3.

C. INTERFACE WITH IBM SYSTEM 360/67

Introduction of the 7-track tapes into IBM System 360/67 at the W.R. Church Computer Center at NPS entailed some further processing of data. In the XDS 9300 a binary number is represented by considering its octal representation, made up of 3-bit digits, whereas in the IBM 360 the same number is represented by hexadecimal, made up of 4-bit digits. As an example of this difference, a decimal number $(668)_{10}$ is $(1234)_{7}$ octal and $(29C)_{16}$ hexadecimal. It was necessary, then, to write a FORTRAN program to read each record in a file from the 7-track tape, make the conversion from octal to hexadecimal, store the data in hexadecimal form on a temporary storage disk, and to print out a portion of each record in decimal form. A listing of the program is included as Appendix B. The program uses the pre-compiled assembler language subroutine FORM to convert the data to its hexadecimal representation. Figure 3 shows the flow of information through the IBM System 360/67 computer in this validation.

The printed output from the conversion program is representative of the data in its original measurement form multiplied by a gain factor of 50 in the A-to-D conversion process. At this point in the process the numbers printed out were compared with the signals recorded on the Brush Recorder during A-to-D conversion to insure that the accuracy of the data had been preserved. This was indeed the case with the exception of some small amplitude,

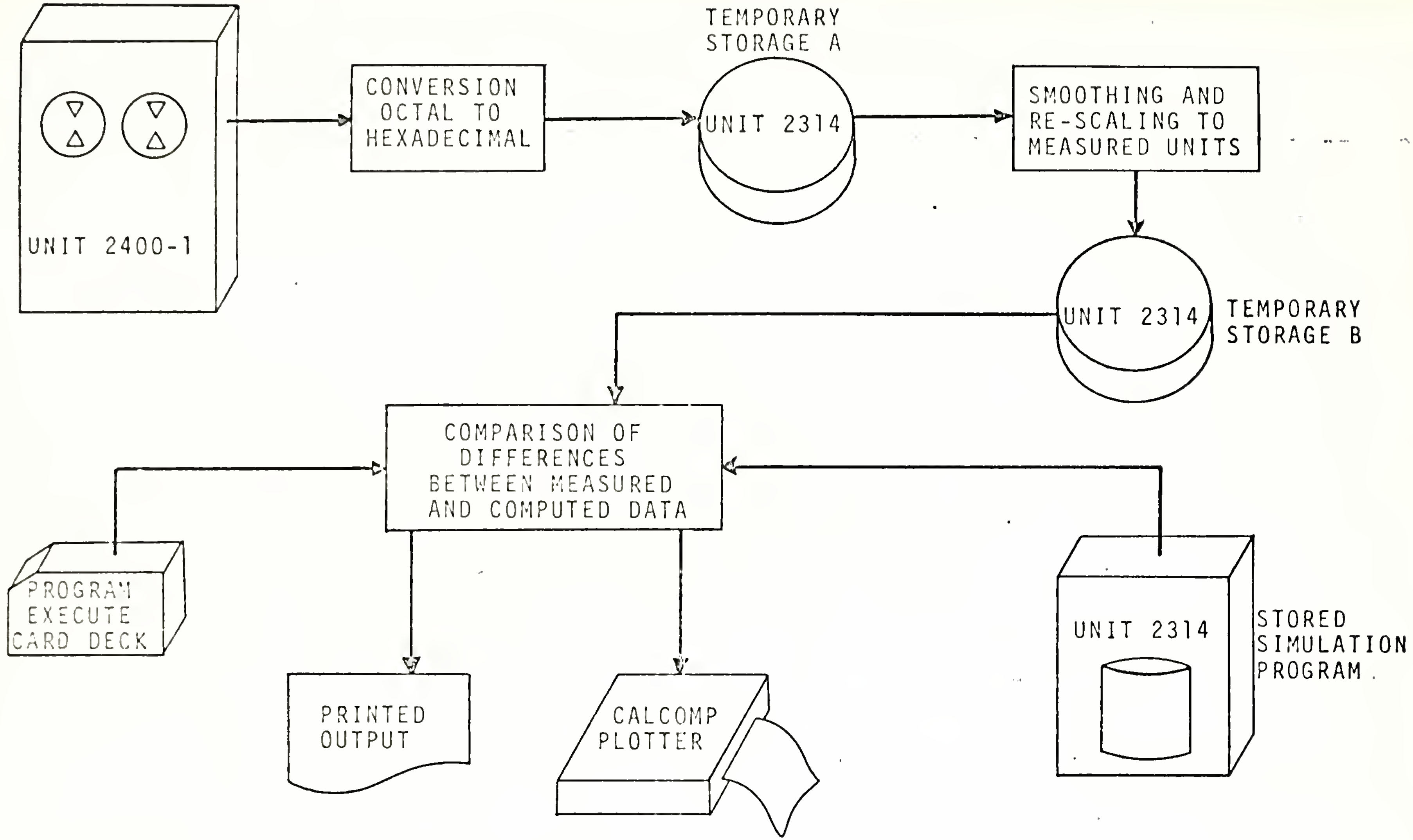


FIGURE 3. INFORMATION FLOW THROUGH IBM SYSTEM 360/67

high-frequency fluctuations in the digitally reproduced data which did not appear to be present on the Brush Recorder tapes.

The final phase in data processing was to recover the measured data into the units which correspond with the measurement, i.e. pounds of thrust, degrees of pitch angle, knots of velocity, etc. to facilitate direct comparison in the XR-3 simulation model. To accomplish the recovery of data it was necessary to read the converted data from the temporary storage disk, multiply each set of variables by the inverse of the appropriate scale factor used when recording the data on the testcraft, divide out the gain factor of 50 introduced in the A-to-D process, and store the result on a temporary disk for later access by the XR-3 simulation program.

The height measurement made on the XR-3 during data collection runs does not have an immediately available counterpart in the XR-3 program as indicated in Table 1. A very close approximation to draft can be obtained from the height measurement through suitable scaling, however. The calculation of draft in the XR-3 simulation program is made with reference to draft at the center of gravity of the craft. The height measurement is made at a sensor located 191" forward of the reference center of gravity and 6.375" down from the deck level of the test craft (see Figure 1). The measurement of the height of the sensor above the surface of the water will vary as the

craft pitches up and down at the bow. Since pitch angles are, for the normal operating range, less than five degrees, the small angle approximation may be made for the sine of pitch angle. The draft measurement is synthesized by solution of the following relationship:

$$DRAFT = SH + Z - HT + D*SIN\theta$$

where DRAFT = Water height above keel at the center of gravity

SH = Height of sensor above CG = 7.145"

Z = Distance of CG above keel = 30.48"

HT = Measured height of sensor above water surface

D = Distance of height sensor forward of CG = 191"

$D*SIN\theta$ = Variation of measured height due to pitch angle.

A FORTRAN program was written to accomplish the process of re-scaling to measured units and to provide the close approximation to a draft measurement. A time index was also included in the program to facilitate plotting each variable versus time on a CALCOMP plotter. This feature proved to be valuable as a final step in the process because each variable could be closely examined in graphical form prior to accessing the data by the XR-3 simulation program. It was at this point in the procedure that some difficulties were encountered in early attempts at signal processing, as described in the following section.

D. FREQUENCY COMPONENTS ENCOUNTERED IN DATA

Examination of the CALCOMP plots of the digitally reproduced measurement data revealed significant small

amplitude, high frequency noise impressed on the measurement signal. The noisy signal was not immediately evident on the Brush Recorder tape, however on close inspection it was apparent that some noise was passing through the filters. Figures 4,5,6, and 7 are representative CALCOMP plots of processed measurement data of thrust, pitch angle, velocity and plenum pressure respectively. These graphs are included to show the effect of noise on the processed data.

The source of noise present in the measurement data has not been established and can only be rationalized to be a combination of several possible environmental factors. First of these is vibration aboard the XR-3 testcraft from the twin engine propulsion system and fan drive system. Second, the testcraft is rarely run in completely glass-calm water conditions at the test site. Surface disturbances are nearly always present due to numerous pleasure craft, wind-induced ripples on the lake, and occasionally, the wake of the testcraft itself. Such disturbances can have a pronounced effect on the motion of the XR-3. Finally, 60-cycle power supply interference can be assumed to be present due to the antenna effect of unshielded wiring both in the testcraft and in the NPS Computer Laboratory.

1. Elimination of Unwanted High Frequencies

Recognizing that the noisy data rendered it all but unusable for any meaningful attempt at validation, methods were sought to eliminate the noise and to obtain a "clean"

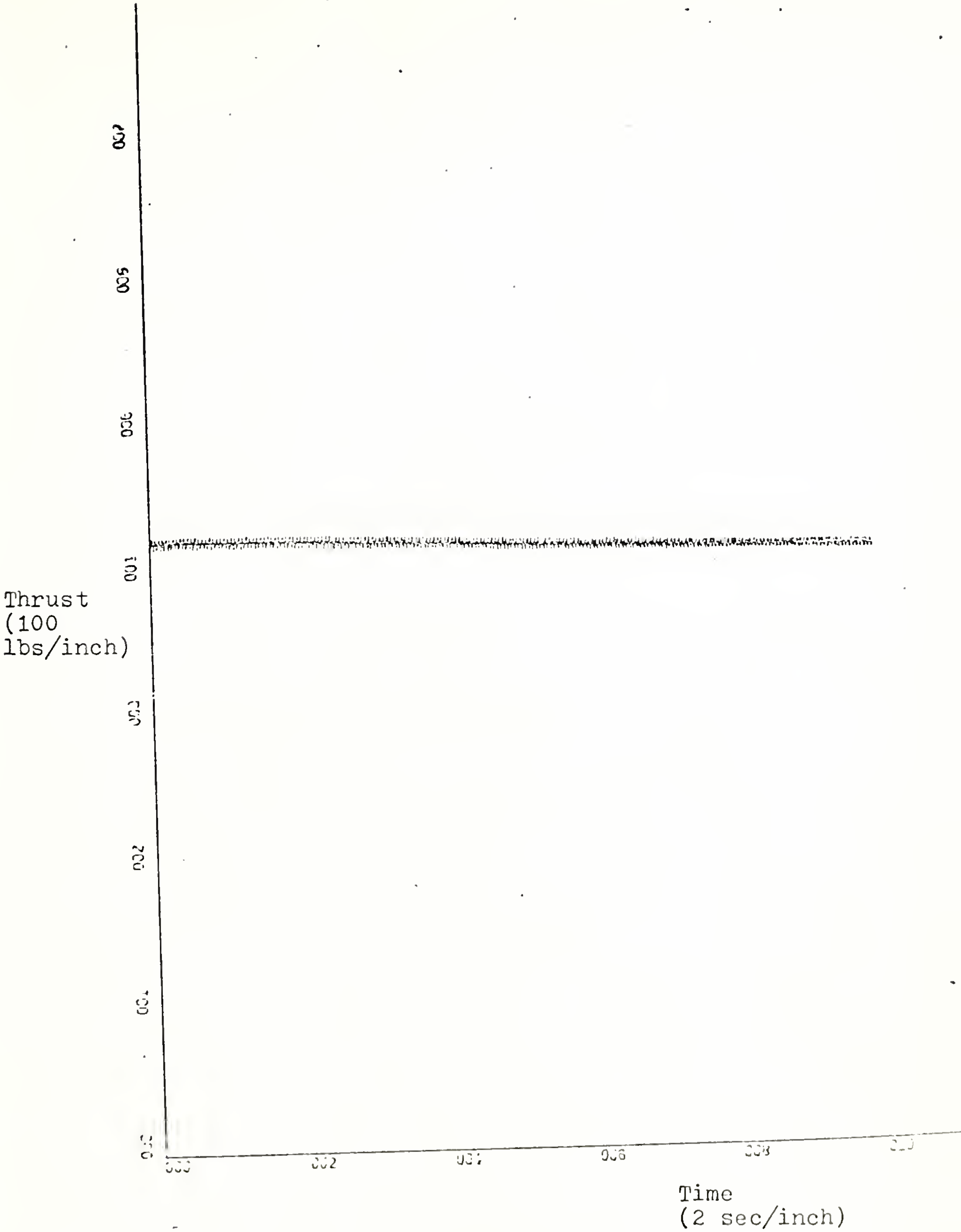


Figure 4. Plot of Thrust Vs. Time

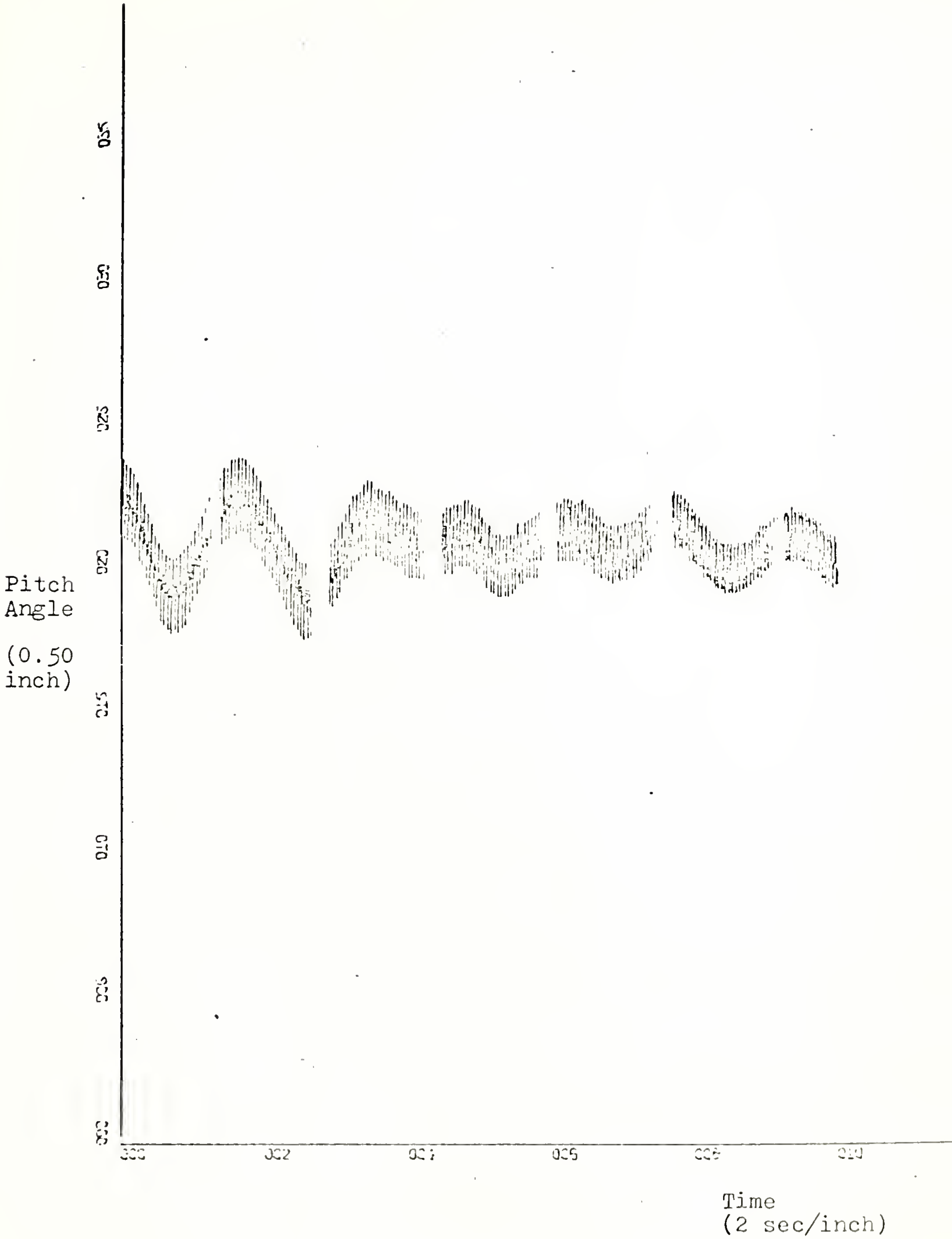


Figure 5. Plot of Pitch Angle Vs. Time

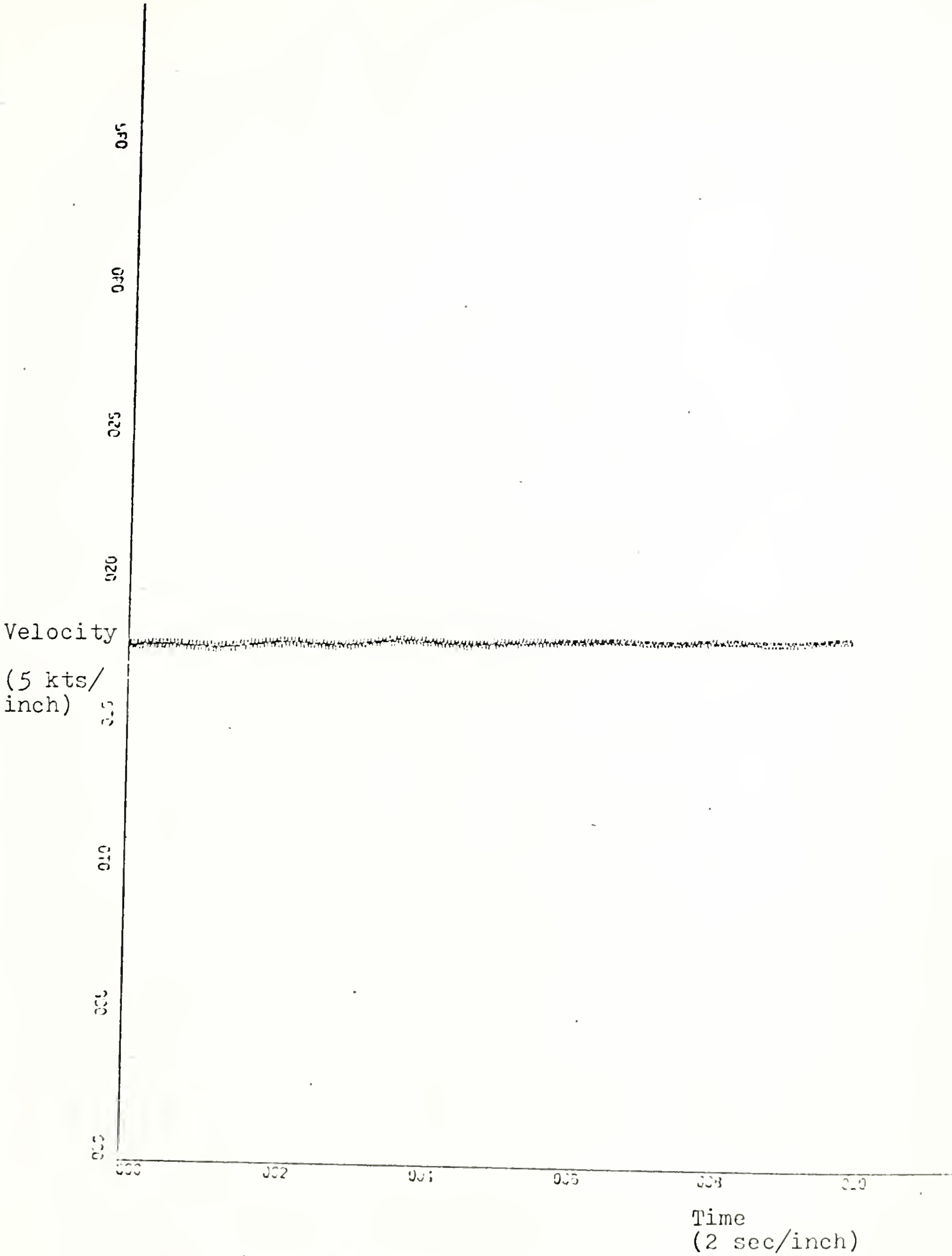


Figure 6. Plot of Velocity Vs. Time

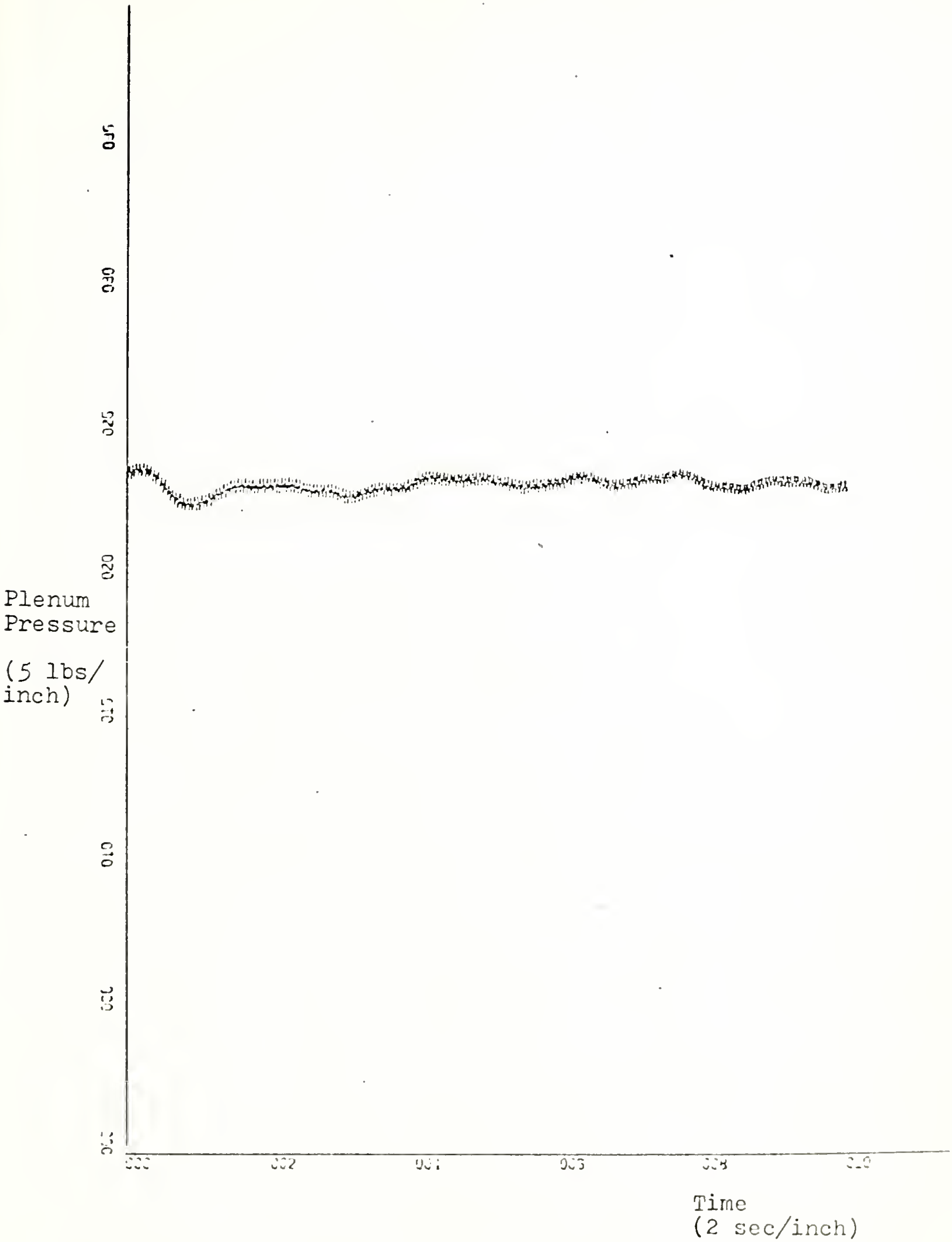


Figure 7. Plot of Plenum Pressure Vs. Time

signal and, at the same time, to preserve the validity of the actual measurement. The first effort was to filter at a lower frequency. The result was displeasing for, while the magnitude of the impressed noise was somewhat reduced, lower frequency oscillations which represented actual craft motions were nearly eliminated. The next possible solution was to employ two filters in cascade, thereby obtaining a low-pass filter with 48 db per octave attenuation above the cut off frequency. This procedure accomplished the main objective of removing virtually all the noise and passing only signals representative of the craft motions. The disadvantage of this method was that only three instead of six measurements were now possible because of the hardware limitations and synchronization became an insurmountable problem.

The third possibility of removing the noise from the measurement signal was to proceed on the assumption that the noise was gaussian and white and to employ a smoothing or time-averaging technique. The technique employed was to process the data as originally set forth and then, in the final scaling process on the digital computer, to obtain a moving time average. The algorithm implemented on the digital computer was to calculate the sum of the first ten points in the time history of a measured parameter, divide the sum by ten and call the resultant figure the first point in the time history. Next, a counter was incremented by five and the process was repeated to obtain

the next point. In this manner the data is reduced from 200 samples per second to 20 samples per second. The overlapping smoothing technique tends to preserve low frequencies. By comparison there were insignificant differences (less than two per cent) between this smoothing process and the aforementioned cascade filtering. The advantage realized was a more economical use of available equipment (filters) and a great savings in time required for data processing.

A listing of the program used for re-scaling the data to its original measurement form and including the smoothing routine is included as Appendix C. Figures 8, 9, 10 and 11 are CALCOMP plots of smoothed measurement data of thrust, pitch angle, velocity and plenum pressure respectively. By comparison with the measurement data plotted in Figures 4,5,6, and 7, it is evident that the moving time average smoothing technique has preserved the measurement information of value and has eliminated the impressed noise.

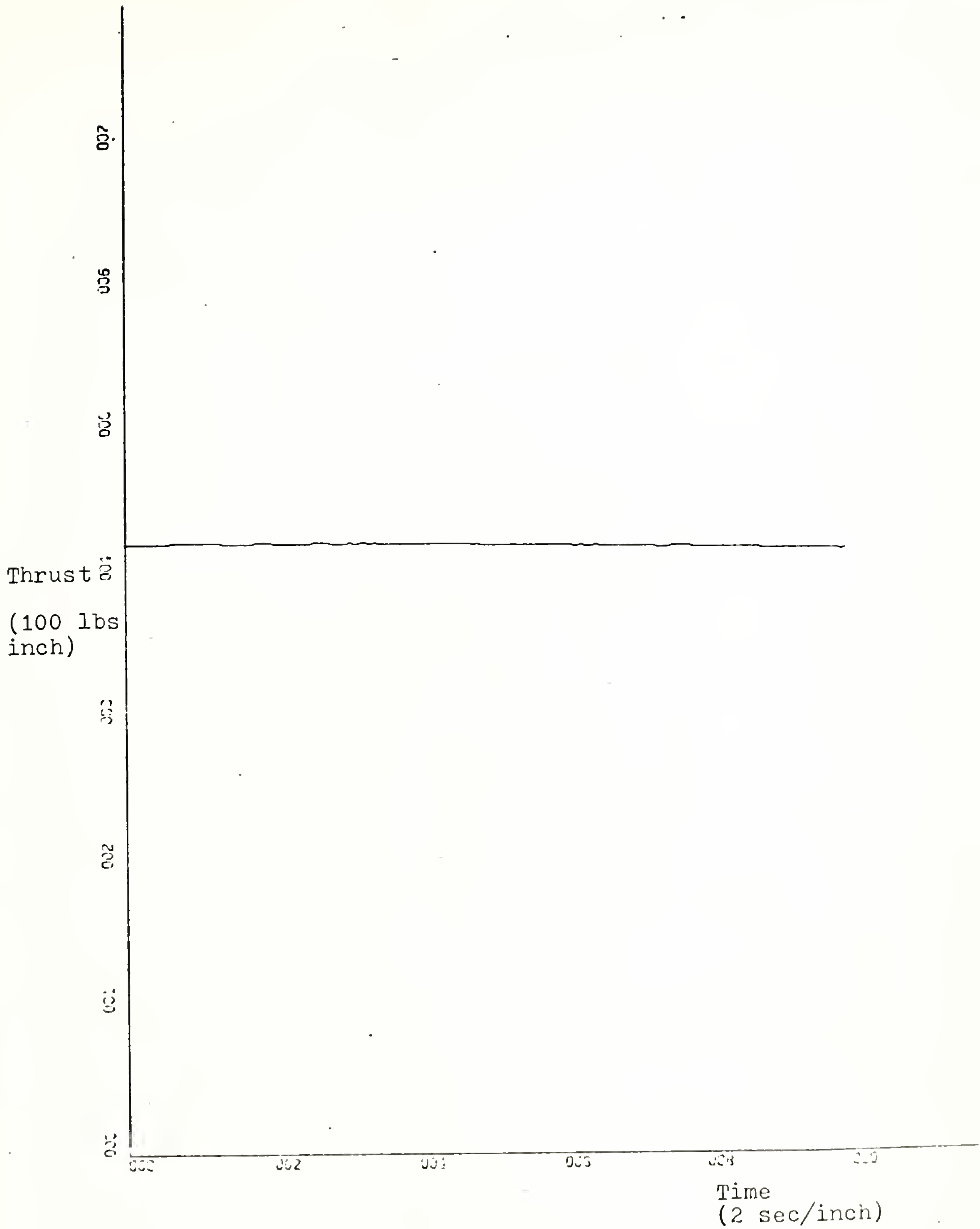


Figure 8. Plot of Thrust Vs. Time After Smoothing

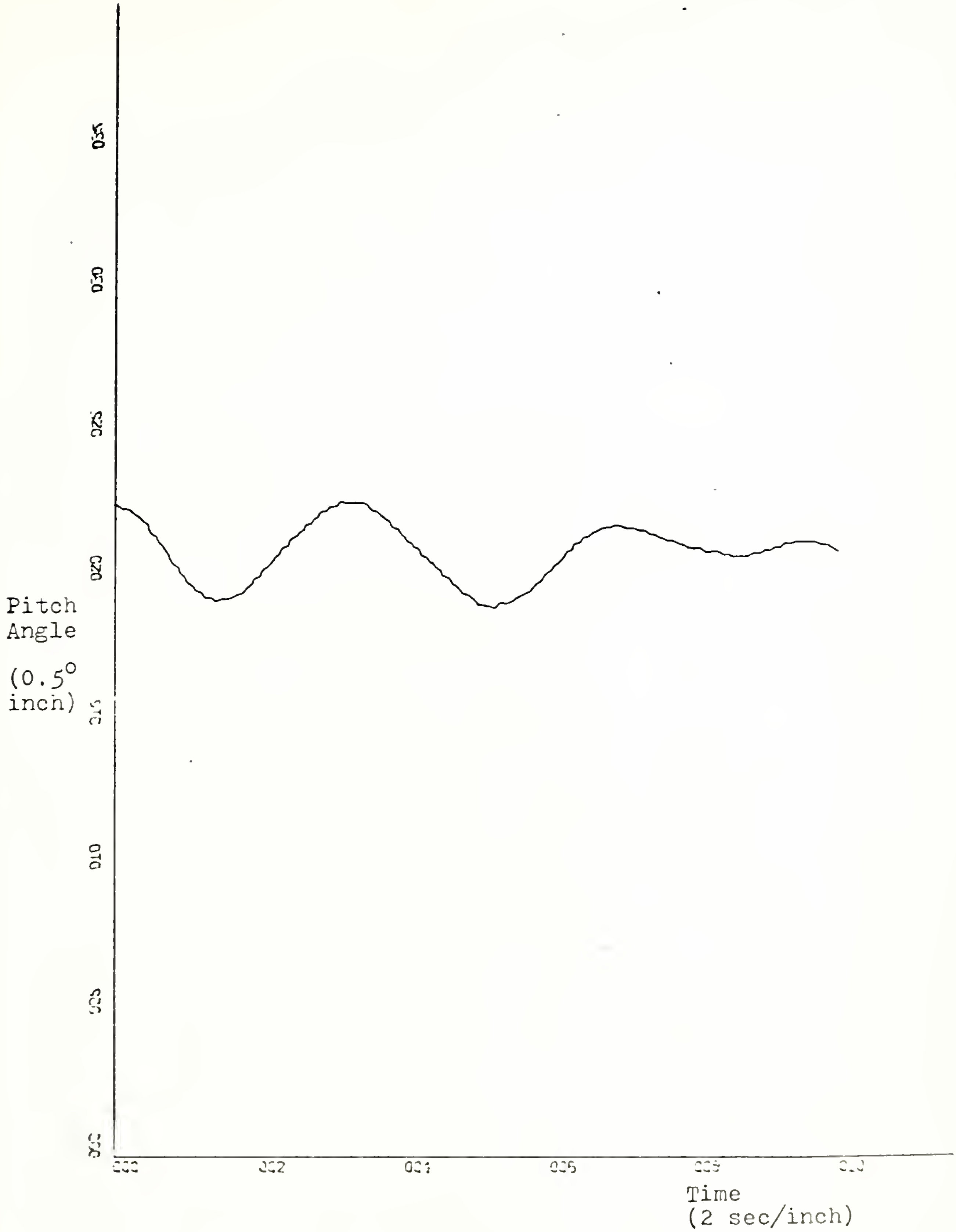


Figure 9. Plot of Pitch Angle Vs. Time After Smoothing

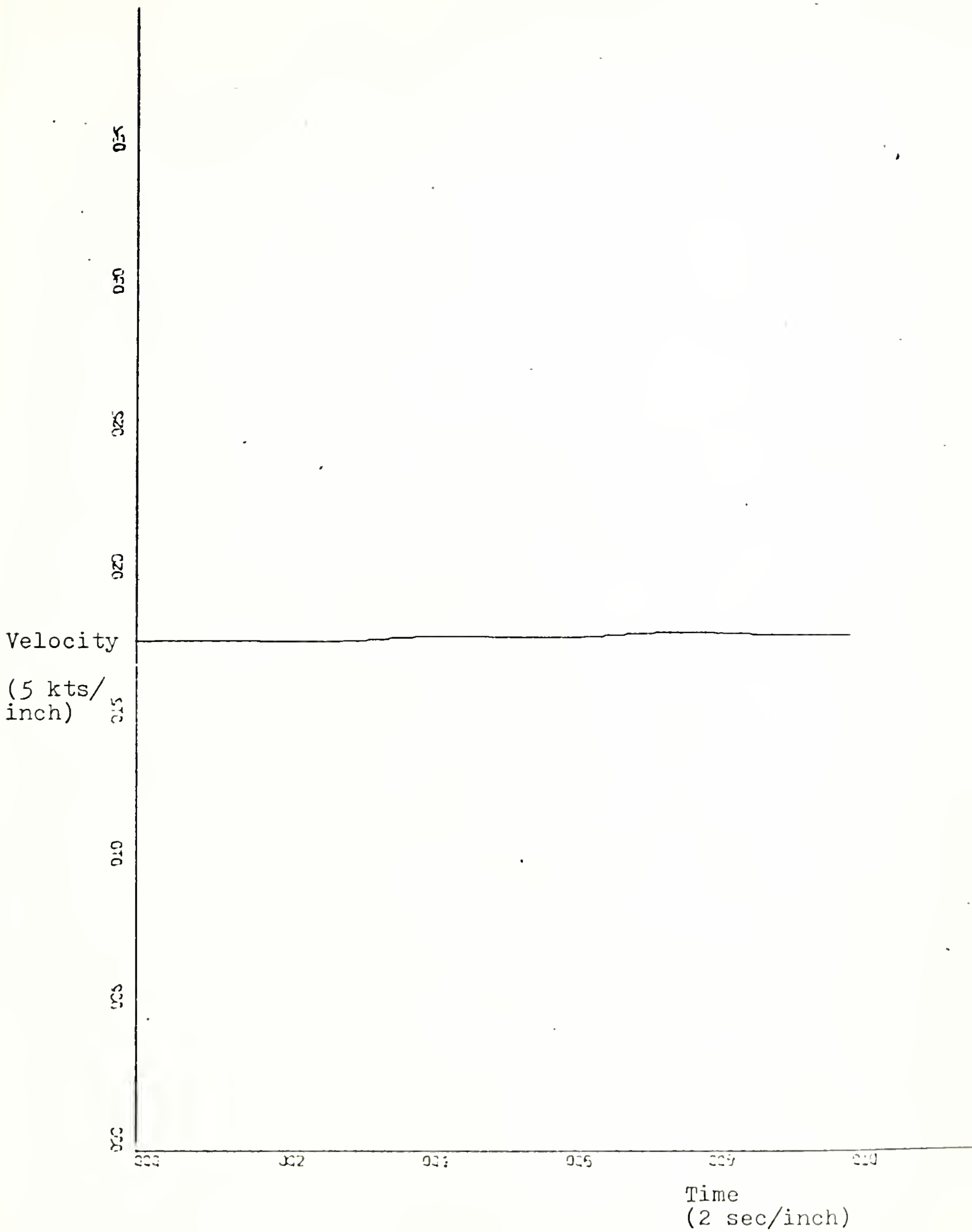


Figure 10. Plot of Velocity Vs. Time After Smoothing

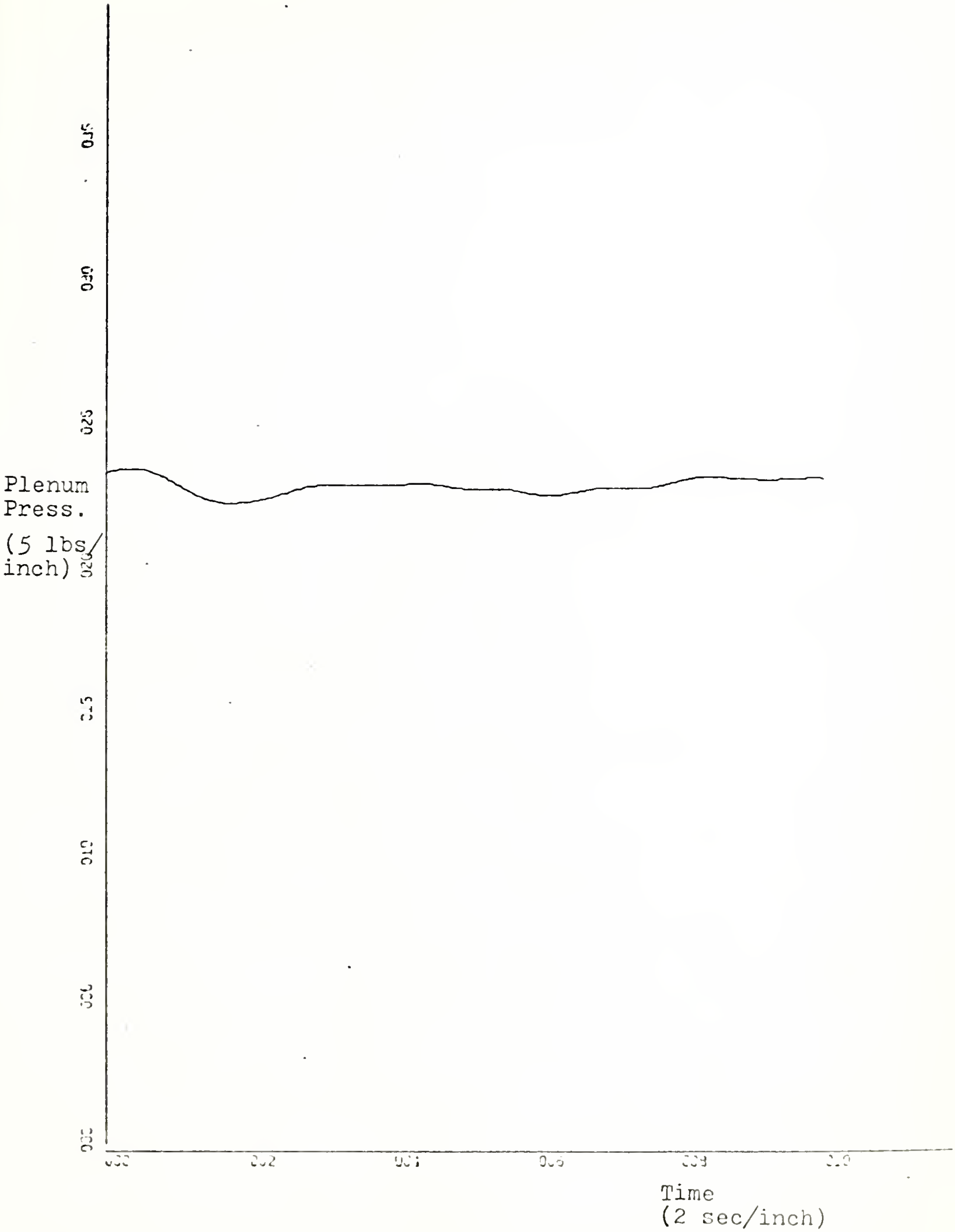


Figure 11. Plot of Plenum Pressure Vs. Time After Smoothing

IV. INTERFACE WITH XR-3 SIMULATION PROGRAM

A. LOGIC ADDITIONS TO SIMULATION PROGRAM

The XR-3 Loads and Motions simulation program was modified to the extent that, on execution of the program, the processed measurement data could be accessed if desired and on-line computation of differences could be made. The output from the program would then be a time history of the differences between computed and measured parameters. This entailed the addition of another option switch in the input data which is read by subroutine INCON. If the option switch is on or equal to one, subroutine INCON reads the measurement data into core from the disk where the data is stored. If the option switch is off or equal to zero, no reading of measurement data occurs.

It was decided to restrict validation or data comparison runs to ten seconds of simulation time because of core storage and computer time allocations. This decision set the size of the measurement data arrays at 200 elements. A COMMON block was added in subroutine INCON and RHS so that the measurement data could be available in subroutine RHS. As with other COMMON blocks, the feature to zero all elements of the added COMMON block was included in the main program. The following XR-3 test craft variables and the associated simulation names were selected for validation:

Total Thrust	THST
Velocity	VEL
Height	DRFT
Pitch Angle	THETAR
Roll Angle	DPHI
Plenum Pressure	PBAR
Yaw Angle	DEPSI
Rudder Angle	DELRS
Vertical Acceleration	ACCEL(3)
Yaw Rate	RDEG

The necessary logic was added in subroutine RHS such that the appropriate element with respect to time in the measurement array could be compared with the computed variable and a difference formed. The resultant differences were written into a temporary file for output by subroutine COLFIL after the simulation run reached the specified final time of ten seconds. Computation of differences was arranged such that positive values indicated that the computed parameter was greater than the measured parameter for any given sample time. The print interval for output purposes was set at one point every 0.05 seconds to correspond with the measurement data rate.

B. METHOD OF THRUST INPUT AND OBSERVED DIFFERENCES

The first step in the validation process was to initialize the mathematical model at a particular operating point with the expected trim conditions of velocity, pitch

angle, draft, and plenum pressure. The feature exists in the simulation program to provide a thrust map of up to 25 points. Additional logic was included to select every fourth point in the 200 element array of measurement thrust to be applied to the mathematical model as an input variable and a ten second simulation proceeded to produce computed velocity. All other measurement parameters except velocity were ignored during initialization runs. The difference between computed velocity and measured velocity was taken as an output from the model. Following initialization of the model, another computer simulation was made using computed thrust as necessary to hold velocity constant. All measurement data for the run was read in and differences between computed and measured parameters were taken as output from the computer model.

For validation studies using measurement data with very nearly constant thrust applied on the testcraft, the foregoing procedure could be shortened by including only the last phase. Provided that the initial trim conditions were known for input to the computer model, no loss of accuracy resulted in the process of validation.

C. METHOD OF RUDDER AND THRUST INPUT TO THE MODEL

Similar to the thrust map provision is a provision for a rudder map input to the computer model. Up to 25 points may be included in a rudder map and the necessary logic was included in the program to select every fourth

point in the 200 element array of measured rudder angle for input to the mathematical model. The objective here was to apply the same inputs to the mathematical model as were used on the testcraft along with the initial trim conditions for each case. This method proved to be of little real value because a steering cable failure to the port propulsion engine on the testcraft rendered the rudder position measurement unreliable. To substitute for direct steering control on the port engine, the starboard engine was mechanically linked to the port engine by means of a long, rigid bar. Thus, the starboard engine controls the port engine in the rudder position and alignment of thrust vectors on the testcraft. Virtually all precision in rudder angle measurement is lost due to unavoidable play in the linkage and lack of direct steering control over both engines.

Validation studies involving turning maneuvers with the XR-3 testcraft and the mathematical model were carried out by matching yaw angles and yaw rates for a spectrum of operating conditions and by noting the differences between computed and measured rudder angles for each operating condition.

V. PROGRAM SUBROUTINE MODIFICATIONS

A. REPLACEMENT OF CENTER OF PRESSURE CURVE

Newman and Poole reported in Ref. 3 the existence of a pressure wave and the attendant gradual deformation of the surface of the water due to a moving pressure distribution with regard to CAB type hulls. NSRDC found that, in towing tank tests of CAB scale models, the craft tends to pivot in pitch motion about a point approximately 75-80% of the wetted length of the sidewall forward of the transom (Ref. 4). Based on these findings, it has been postulated that the center of volume within the plenum chamber moves aft with increasing speed.

Accordingly, the calculations made in the XR-3 simulation program to account for a shifting center of pressure were removed from subroutines RHS and WAVES. Computations were inserted in the program to account for the addition of an incremental wedge of volume to plenum volume which varies with craft speed. What follows is a description by subroutine of changes which were entered and the basis for the inclusion of the changes.

1. Subroutine INCON

A previously unused space on the input data card number 00701 was used to read in the location of the pressure wave pivot point, measured in feet and tenths of feet forward of the transom. The statement XPWV = TEMP(3)

was inserted in Block 07 in the INCON subroutine. A value of 17.2 ft. was used for XPWV in this study.

The relationship for wave drag produced by the movement of the pressure bubble over the water surface is, from Ref. 5:

$$X_{bubble} = -(P_b - P_a)\, C_F \frac{4W}{\rho g l} .$$

The same formula is mechanized in the simulation program as:

FXPWAV = -PBBAR * CF * PWVCON

where CF = 0.37 / ((U/FNCON) **1.5655981) and PWVCON = 4.0* WEIGHT /(RHO *G* XLBW). The coefficient 0.37 and the exponent 1.5655981 are correct for the length to beam ratio of the XR-3 as determined from Ref. 3. The variable name WATSLP is given to the slope of the water surface inside the plenum and is calculated as:

WATSLP = PBBAR * CF * PWVCON / WEIGHT.

WATSLP represents the angle at the apex of the wedge-shaped increment of volume to be added to plenum volume. Since the ratio of PBBAR*CF*PWVCON (i.e. bubble drag) to WEIGHT will always result in a small number for speeds above hump speed, the small angle approximation may be used without loss of accuracy. Therefore, the volume increment due to the water slope to be added to plenum volume is given by

0.5 * WATSLP * XL * AB

where XL is the length of the plenum chamber and AB is the area of the plenum at the water surface. The quantity 0.5 * WATSLP * XL is then the vertical cross sectional area of the wedge-shaped volume increment.

Two other calculations were added in subroutine INCON for use in other subroutines during the simulation. The distance from XPWV from the forward extremity of the plenum is calculated as

XLXPWV = XLBW - XPWV

where XLBW represents the plenum length at the water surface. Also the distance between the craft center of gravity and XPWV is calculated as

XPWVXS = XPWV - XS

where XS is the longitudinal center of gravity measured in feet forward of the transom.

2. Subroutine BOWSL

In the computation for ELSKI, the wetted lenght of the bow seal, an incremental length is added to the wetted length of the seal. The increase is due to the water slope and is calculated simply as

XLXPWV * WATSLP

and added to the existing expression for ELSKI.

3. Subroutine STNSL

Similar to the wetted length correction in BOWSL, an incremental correction is computed and applied in subroutine STNSL. Here the correction is opposite in sign and is applied to the wetted length of the stern seal, ELSKI as

-XPWV * WATSLP.

4. Subroutine RHS

The water slope correction is computed in subroutine RHS as a continuous variable over the period of any given

simulation time. Recognizing that WATSLP is calculated as the ratio of bubble wave drag to total craft weight and that bubble wave drag varies with velocity and plenum pressure, WATSLP must be considered as a continuous variable. Accordingly, the following computation was introduced in RHS:

WATSLP = -FXPWAV / WEIGHT

The plenum volume is corrected for the water slope on each pass through subroutine RHS by the addition of a volume increment as in subroutine INCON:

+ .5 * WATSLP * XL * AB.

5. Subroutine SIDEWL

The addition of a wedge-shaped volume to plenum volume required an alteration to the computation of the drafts along the sidewalls inside the plenum chamber. Displacement of water is calculated as

PBHEAD = PBAR / (RHO * G)

where PBAR is the instantaneous plenum pressure. The immersion depth on the inside of each sidewall section is calculated in the relation

DDIN = DD - WATSLP * (XPWVXS - XX(,K))

where DD represents the uncorrected water level height above the keel along each sidewall section and XX(J,K) is the distance from the transom of the center of each of the eleven stations along the sidewall.

A revision of the method of computing sidewall gaps at any given station was also necessary. A logical IF

statement was included to test each value of DDIN after it was computed. If DDIN is less than zero another logical IF statement must be tested to determine if a gap exists between the bottom of the sidewall (keel) and the water surface. If DSW (J,K), or the actual sidewall immersion depth, is less than PBHEAD, then a gap for that sidewall section is computed as

GAP (J,K) = - DDIN * (1.-(DSW(J,K))/ PBHEAD)

where the computed GAP for any given station is in units of feet.

Further background information of use in this study is contained in References 6-11.

B. THRUST VECTORS IN SUBROUTINE PROP

The propulsion system in the XR-3 consists of two forty horsepower outboard motors. The port screw has a right hand rotation and the starboard screw has a left hand rotation. Steering control is obtained by rotation of the motors in the horizontal plane. Thus, there is some rudder effect from the foil-shaped lower motor housings, however the predominant turning moment results from re-alignment of the propulsion thrust vectors when rudder angle is applied. In addition to the main thrust vector, directed in the line of craft motion for zero rudder angle, there exists a side thrust vector for each propeller. Since the propellers are counter-rotating, and due to the direction of rotation of each propeller, the side thrust vectors are

oppositely directed and, for zero rudder angle, are directed inboard.

The assumption is made that both port and starboard propulsion motors rotate together through the same rudder angle. It is also assumed that the magnitude of the main and side thrust vectors are independent of rudder angle.

The propeller thrust diagram shown in Figure 12 was used to verify and correct the force and moment equations for the XR-3 propulsion system. The figure displays the main thrust and side thrust vectors for both propellers positioned with right (positive) rudder angle δ, positive pitch angle θ, and negative (port side down) roll angle ϕ. The thrust vectors act through the port and starboard propeller positions, XP, YP, and ZP from the craft center of gravity in the Body Reference coordinate system.

The forces associated with the propeller alignment of Figure 1 are shown vectorially in Figure 13 in the Local Level coordinate system after having been corrected for pitch and roll angles. The assumption is made that pitch angle and roll angle are limited to small angles such that $\sin\theta \cong \theta$ and $\cos\theta \cong 1$, $\sin\phi \cong \phi$ and $\cos\phi \cong 1$. The following force equations result:

T1X = THST1 * $\cos\delta$

T2X = THST2 * $\cos\delta$

ST1X = - STHST1 * $\sin\delta$

ST2X = STHST2 * $\sin\delta$

T1Y = THST1 * $\sin\delta$

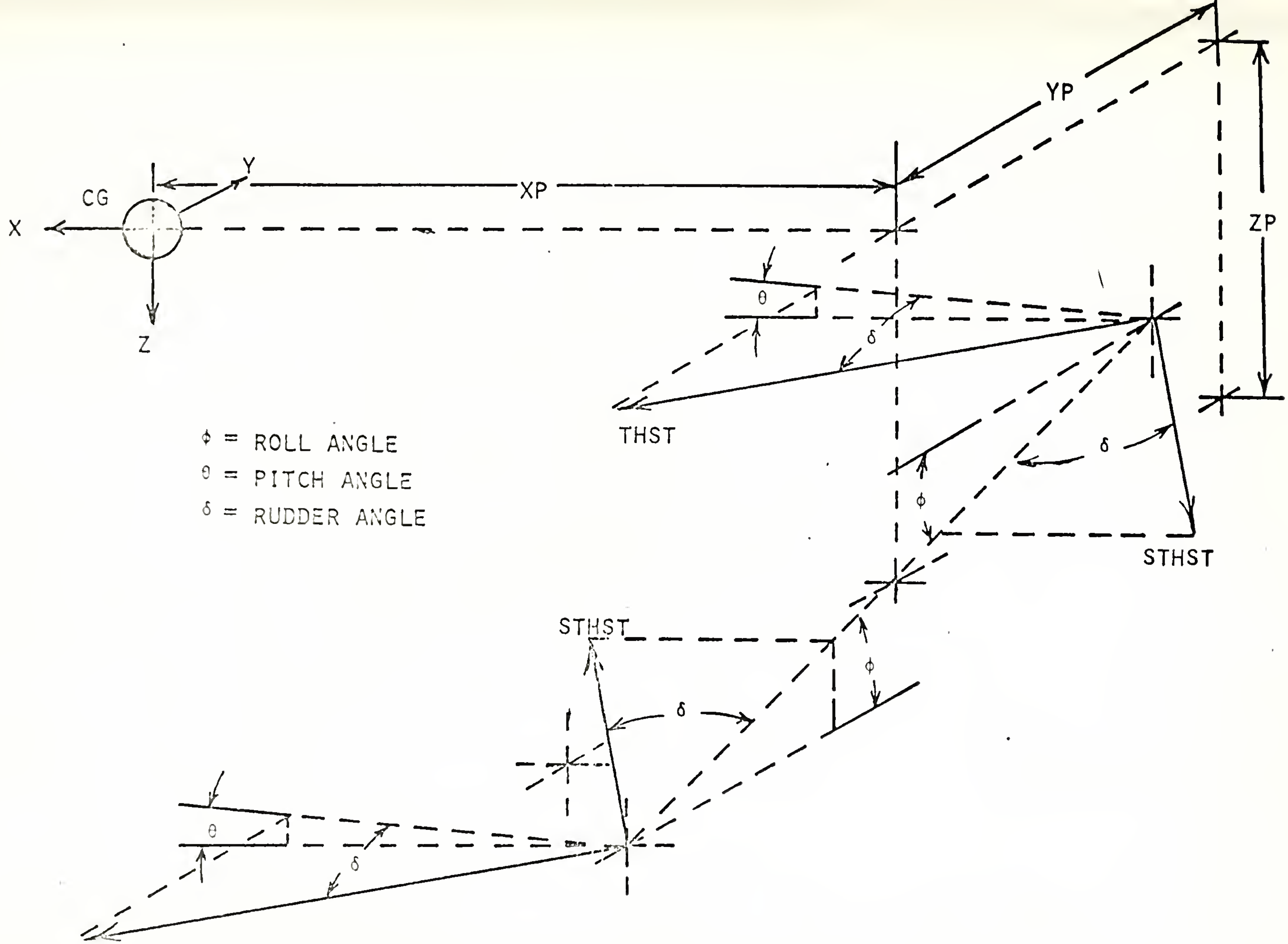


FIGURE 12. PROPELLER THRUST DIAGRAM

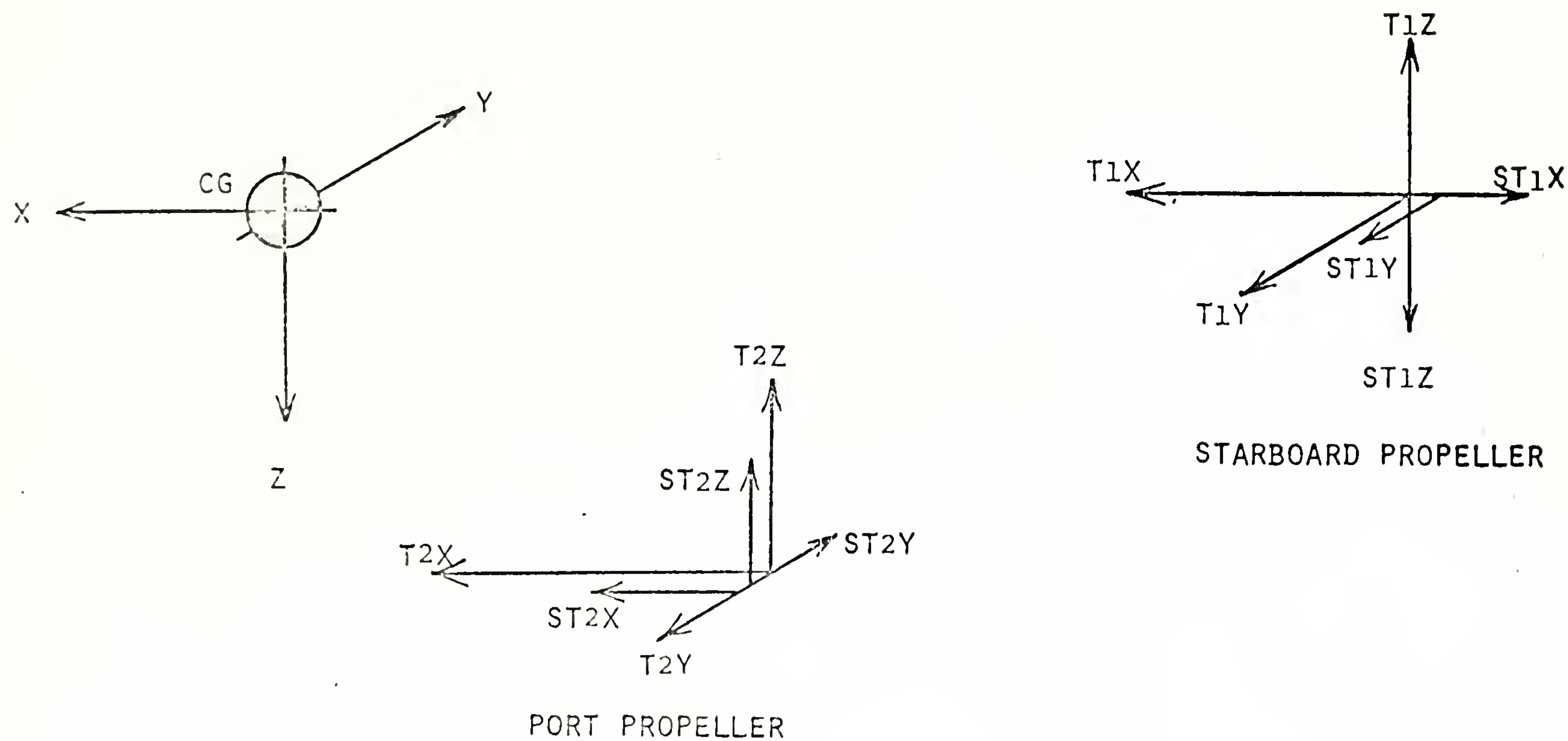


FIGURE 13. VECTOR RESOLUTION OF PROPELLER THRUST FORCES

$T2Y = THST2 * \sin \delta$

$ST1Y = -STHST1 * \cos \delta$

$ST2Y = STHST2 * \cos \delta$

$T1Z = THST1 * \cos\delta * \theta$

$T2Z = THST2 * \cos\delta * \theta$

$ST1Z = STHST1 * \cos\delta * \phi$

$ST2Z = STHST2 * \cos\delta * \phi$

Summing the forces in each of the three corrdinate directions yields the following result:

FXS = T1X - ST1X

FXP = T2X + ST2X

FYS = - T1Y - ST1Y

FYP = T2Y + ST2Y

FZS = - T1Z + ST1Z

FZP = -T2Z - ST2Z

The total force equations are found by summation of the force equations for the port and starboard propulsion systems in each of the three coordinate directions as follows:

FX = FXP + FXS

FY = FYP + FYS

FZ = FZP + FZS

Corrections were made to the force calculations of sub-routine PROP from Reference 1 on the basis of the foregoing discussion.

VI. DISCUSSION AND EVALUATION OF RESULTS

A. OBTAINING AGREEMENT AMONG VARIABLES

Computer simulation runs in the initial stages of this study revealed that plenum pressure was consistently approximately 2.5 psf higher than that recorded on the XR-3 testcraft. Among other variables, thrust on the testcraft was higher than the computer model indicated for any given speed and calculated draft was greater by approximately three inches than the computer model produced. The decision was made to first obtain agreement in plenum pressure between the mathematical model and the testcraft.

The bow seal leakage area input to the mathematical model was listed in Ref. 1 as one of the parameters not known to be exact. The estimate of leakage area was revised from 0.08 sq. ft. to 0.10 sq. ft. This slight alteration had the desired effect of reducing plenum pressure to agree with that measured on the XR-3 testcraft. An unexpected but welcome additional effect was nearly perfect agreement in thrust and draft. Aside from the alterations to the structure of the mathematical model there were no further changes made to the model during the course of the present study.

1. Straight Runs Made with Different C.G. Positions

Validation runs were made interfacing data measured on the XR-3 testcraft with the computer model. The data

presented in Table III is a summary of average differences calculated between time histories of actual craft motions and simulated craft motions for the thrust, velocity, pitch angle, plenum pressure and draft variables. Also included is percentage differences for ease in qualitative comparison. It should be noted that percentage differences in pitch angle measurement appear to be quite large, however in most cases the difference is within the measurement accuracy limitation of $\pm$ 0.5 degree aboard the testcraft.

The total craft weight on the measurement runs listed in Table III was 6270 lbs. In addition to the pilot there was a passenger in the co-pilot position and 250 lbs. of water in plastic containers aboard used for changing the location of the center of gravity of the XR-3.

The data presented in Table IV is a summary of average time history differences with a fixed center of gravity on the XR-3. On the measurement runs listed in Table IV the total craft weight was 6050 lbs. Vertical acceleration was substituted for draft in this series of validation runs.

Figures 14, 15, 16, 17, and 18 are typical of the graphical output from the XR-3 simulation program arranged to display the time histories of differences between computed and measured variables for this study.

2. Runs Made with Turns

A series of eleven validation runs were made with various rudder angles at various speeds. A summary of the

TABLE III. SUMMARY OF DIFFERENCES IN STRAIGHT RUNS-A

Average Differences Between Mathematical Model and Measured Variables are Listed *
(Percentage Differences in Parentheses)

Run	Velocity	CG Location	Thrust	Velocity	Pitch	Plenum Press	Draft
1.	18.2 kts	10.15 ft	-9.0	0.0	-0.65	0.0	-0.6
			(-1.96)	(0)	(-32.5)	(0)	(-4.9)
2.	18.9	10.27	28.0	0.0	-0.8	0.01	-2.1
			(5.9)	(0)	(-38.1)	(.05)	(-15)
3.	16.6	10.03	-17.0	-0.1	-0.2	0.01	0.1
			(-3.9)	(-0.6)	(-10.2)	(.04)	(0.86)
4.	14.3	10.03	-38.0	-0.06	-0.4	0.14	1.2
			(-8.9)	(-0.42)	(-18.6)	(.63)	(11.3)
5.	16.5	10.15	-12.5	-0.1	-0.3	0.0	0.6
			(-3.0)	(-0.6)	(-14.6)	(0)	(5.4)
6.	17.1	10.19	-10.0	0.0	-0.3	0.0	0.6
			(-2.3)	(0)	(-12.0)	(0)	(5.4)
7.	13.9	10.067	3.5	0.0	-0.2	0.01	0.1
			(0.9)	(0)	(-9.52)	(.04)	(.85)

TABLE III. (Continued)

Run	Velocity	CG Location	Thrust	Velocity	Pitch	Plenum Press	Draft
8.	22.0	10.03	32.0	0.05	0.0	-0.7	-1.4
			(6.4)	(0.2)	(0)	(-3.2)	(-11.2)
9.	12.5	10.15	-8.7	0.07	0.2	0.01	0.5
			(-2.3)	(.6)	(9.1)	(.4)	(4.3)
10.	22.4	10.03	38.0	0.0	0.16	0.0	-1.6
			(7.5)	(0)	(9.3)	(0)	(-11.8)

* NOTE: Negative sign indicates computed variable is smaller than measured variable.

Craft total weight was 6270 lbs. in all runs.

TABLE IV. SUMMARY OF DIFFERENCES IN STRAIGHT RUNS-B

Average Differences Between Mathematical Model and Measured Variables are Listed *
(Percentage Differences in Parentheses)

Run	Velocity	CG Location	Thrust	Velocity	Pitch	Plenum Press	Vert Accel
1.	20.6 kts	10.15	29.0	0.0	0.1	0.12	0.01
			(5.6)	(0)	(5.3)	(4.5)	(2.1)
2.	15.2	10.15	-16.0	-0.05	-0.2	0.02	0.008
			(-4.2)	(-0.3)	(-9.1)	(0.1)	(0.9)
3.	14.4	10.15	-11.0	0.03	0.1	0.05	0.004
			(-3.5)	(0.2)	(6.35)	(0.22)	(0.45)
4.	18.7	10.15	21.0	0.01	-0.7	0.01	0.01
			(4.8)	(0.05)	(-22.0)	(0.4)	(2.1)
5.	22.8	10.15	28.0	0.04	-0.3	0.06	0.003
			(6.4)	(0.17)	(-12.0)	(0.3)	(0.4)

* NOTE: Negative sign indicates computed variable is smaller than measured variable

Craft total weight was 6040 lbs. in all runs.

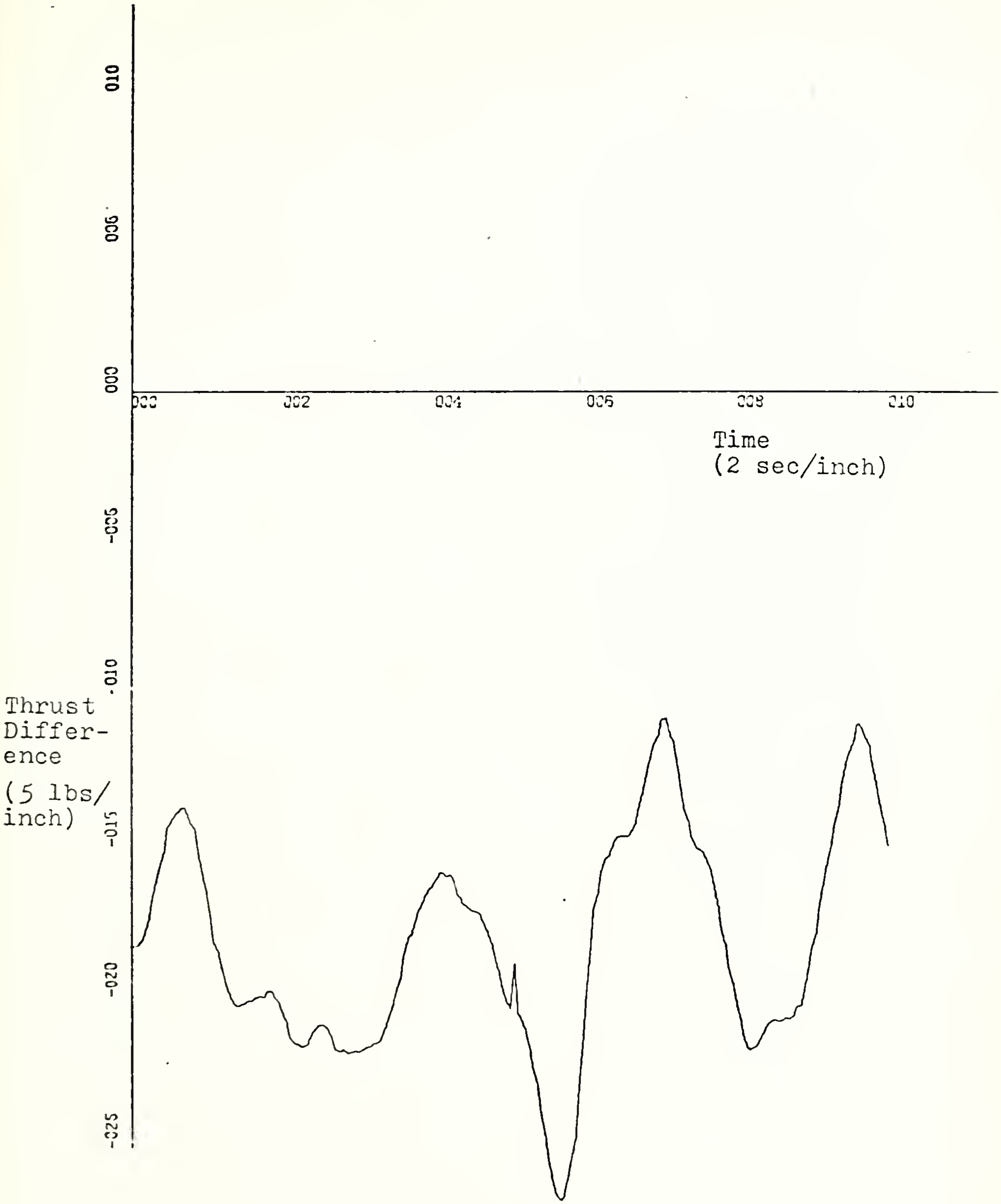


Figure 14. Plot of Thrust Difference Vs. Time

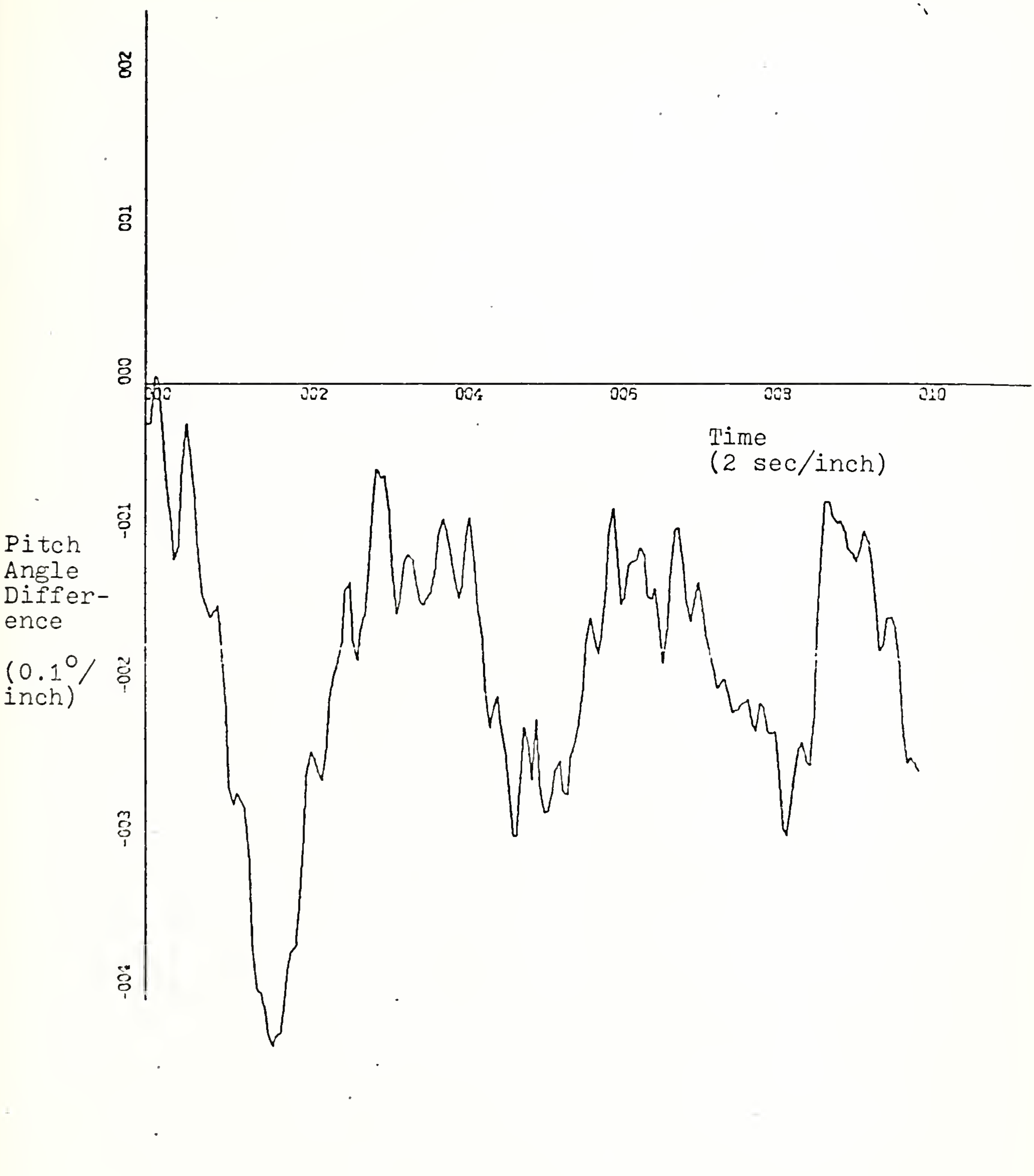


Figure 15. Plot of Pitch Angle Difference Vs. Time

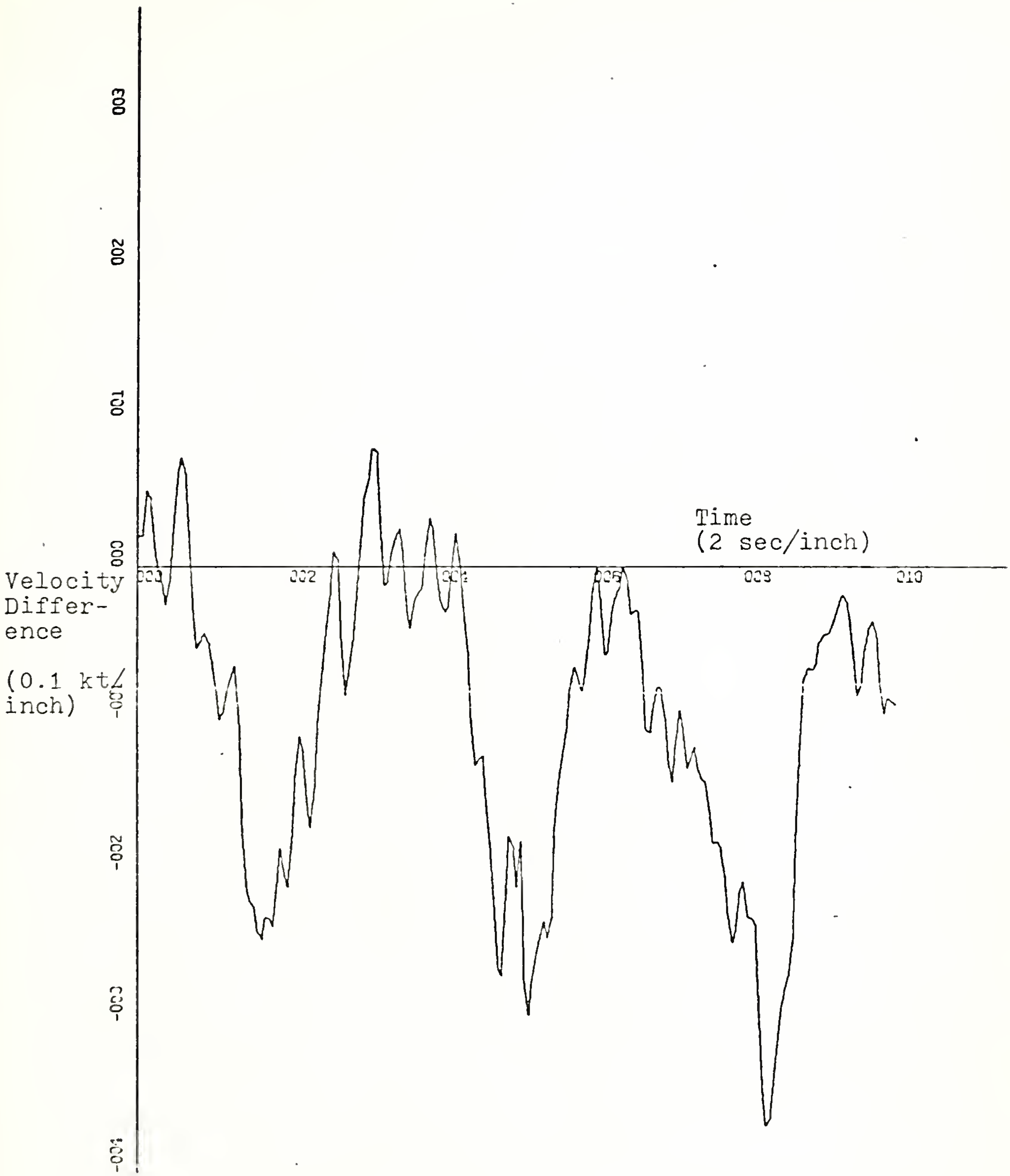


Figure 16. Plot of Velocity Difference Vs. Time

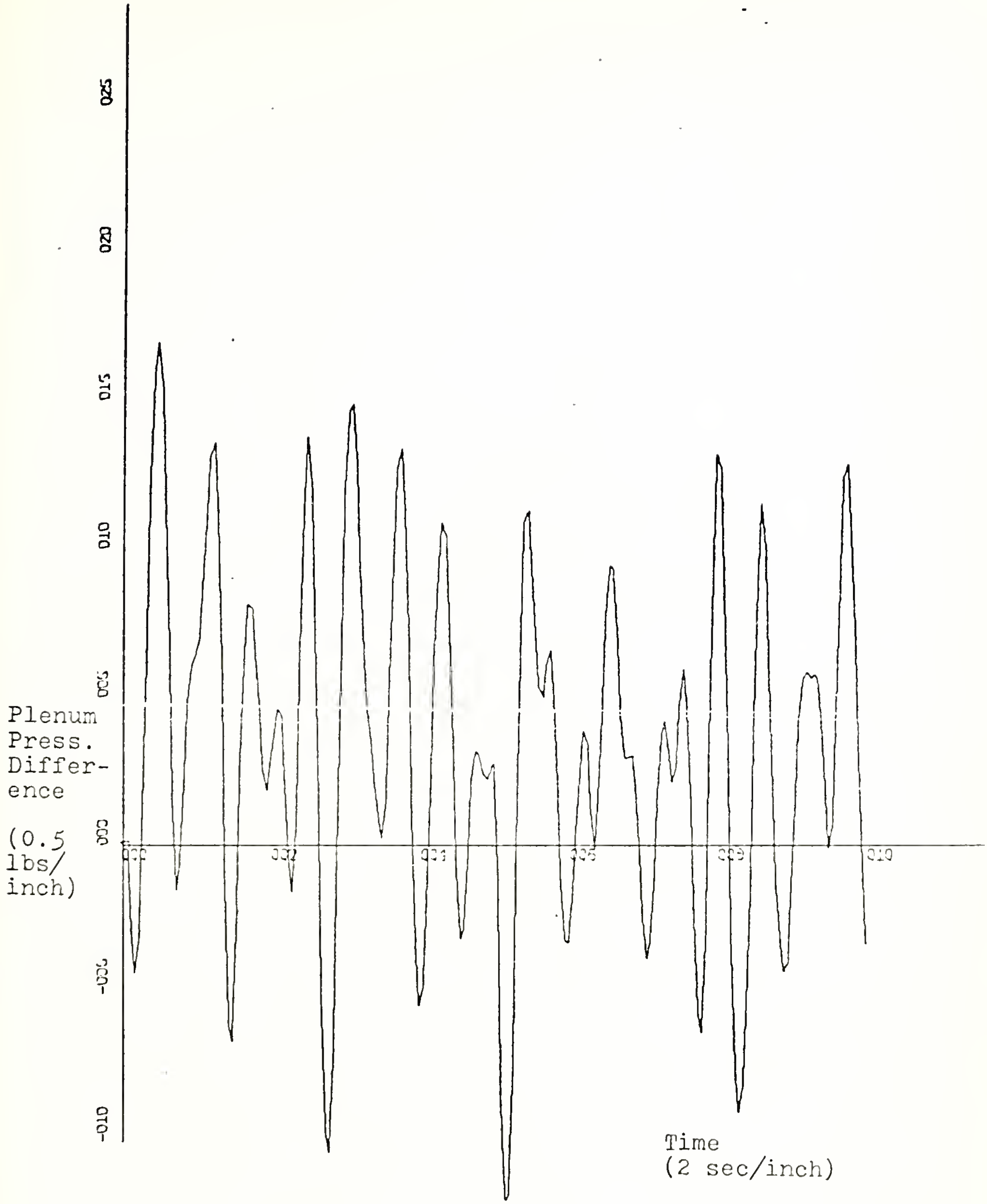


Figure 17. Plot of Plenum Pressure Difference Vs. Time

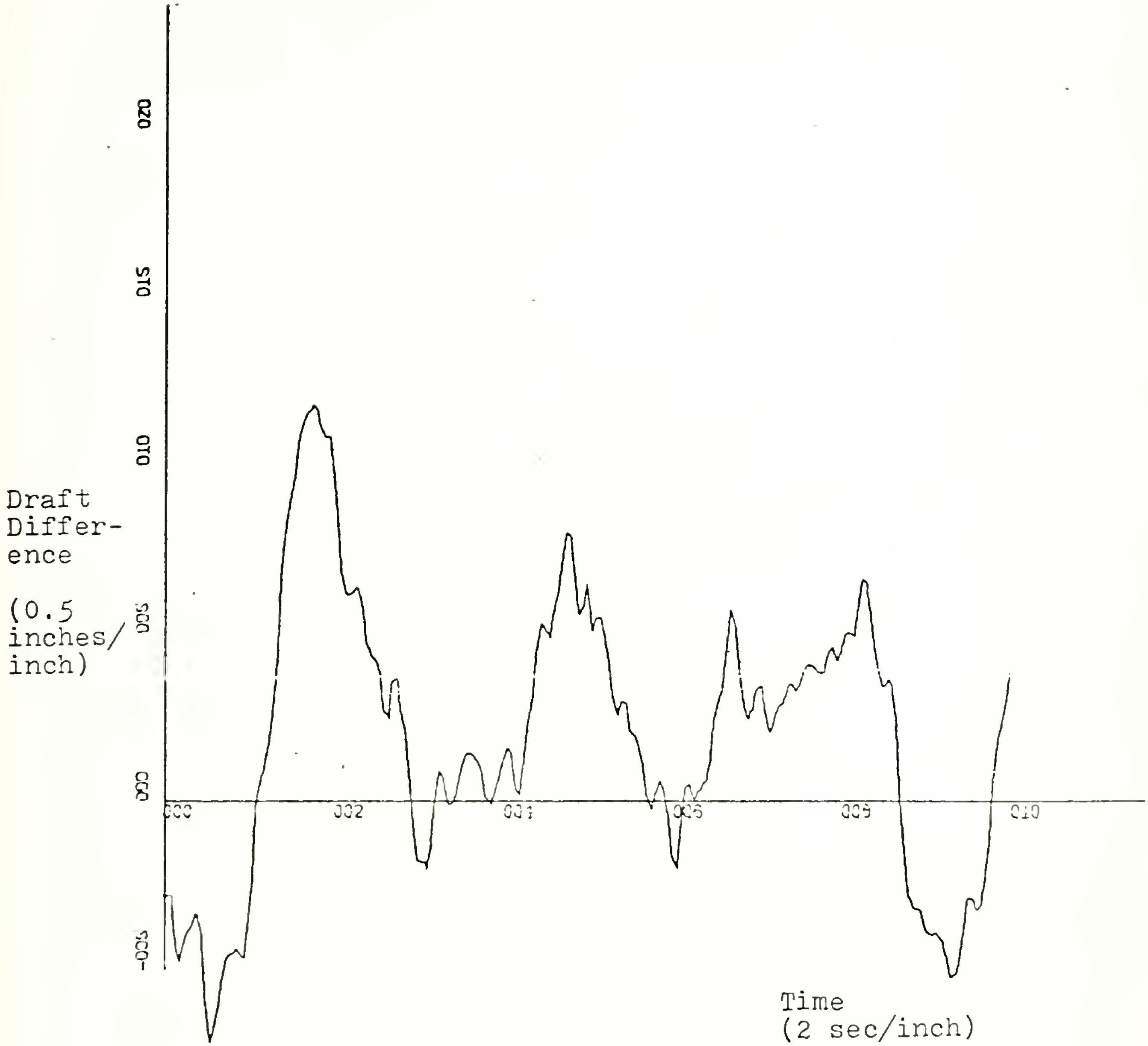


Figure 18. Plot of Draft Difference Vs. Time

average differences calculated between actual craft motions and simulated craft motions during the turning maneuvers for thrust, velocity, pitch angle, roll angle, yaw angle, yaw rate and rudder position variables is presented in Table V.

The relatively large differences in rudder angles in many runs is attributed to the lack of confidence in an accurate rudder position measurement. In the future when direct steering control of both propulsion motors is restored, it is recommended that confirmation be made of the results presented here with regard to rudder position and other variables in turning maneuvers.

Figures 19, 20, 21 and 22 are typical of the graphical output from the XR-3 simulation program which display the time histories of differences in roll angle, yaw angle, yaw rate, and rudder angle, respectively.

B. LIMITED TO CALM WATER CASES

It is recognized that much of the measurement data obtained at the XR-3 test site at Lake San Antonio, California contains small perturbations due to surface disturbances on the lake. No attempt has been made to measure the surface disturbances or their effect on the testcraft. Validation of the mathematical model in the presence of waves of any amplitude and wavelength is suggested as an area for further study.

TABLE V. SUMMARY OF DIFFERENCES IN TURNS

Average Differences Between Mathematical Model and Measured Parameters are Listed *
(Percentage Differences in Parentheses)

Run	Rudder Ang	Velocity	Thrust	Velocity	Pitch	Roll	Yaw	Yaw Rate	Rudder Ang
1.	L 4.5^{o}	19.7 kts	8.5	0.0	-.36	0.03	3.63	0.01	-3.0
			(2.2)	(0)	(-13.4)	(6.5)	(22.7)	(7.0)	(-40)
2.	L 3.0	24.7	12.0	0.0	-.3	0.04	0.5	0.2	-3.0
			(2.5)	(0)	(-13.4)	(8.0)	(4.7)	(21.7)	(-50)
3.	L 3.0	13.7	-17	-0.08	-.7	0.03	-.15	0.0	-8.0
			(-4.9)	(-.3)	(-38.6)	(18.7)	(1.85)	(0)	(-72)
4.	L 3.3	25.1	28.0	0.0	0.05	0.22	1.8	0.3	0.65
			(5.8)	(0)	(2.2)	(38.6)	(15.2)	(29.1)	(16)
5.	L 5.4	16.3	-14.0	0.0	0.09	0.1	-.73	0.09	-1.3
			(-3.9)	(0)	(4.7)	(40)	(-6.2)	(6.4)	(-23)
6.	L 5.8	21.5	-21.0	0.06	0.29	0.23	1.1	0.05	1.4
			(-4.9)	(0.3)	(11.6)	(44)	(6.3)	(2.6)	(31)
7.	R 4.1	19.6	27.0	-0.06	-0.38	-0.1	0.2	0.04	0.7
			(6.8)	(-0.3)	(-20)	(-25.6)	(1.4)	(3.1)	(14.6)

TABLE V. (Continued)

Run	Rudder Ang	Velocity	Thrust	Velocity	Pitch	Roll	Yaw	Yaw Rate	Rudder Ang
8.	R 3.8	19.6	28.0	-0.03	-0.4	-0.13	0.96	0.05	0.44
			(7.1)	(-0.2)	(-20.7)	(-36)	(7.4)	(4.2)	(10.4)
9.	R 6.1	20.2	-18.0	0.05	-0.21	0.15	1.9	0.0	-2.5
			(-4.5)	(0.2)	(-10.5)	(31.2)	(9.2)	(1.8)	(-29)
10.	R 3.9	18.1	16.0	0.0	0.06	0.1	0.62	0.03	-0.9
			(4.2)	(0)	(3.1)	(20)	(5.3)	(2.2)	(-19)
11.	R 5.6	15.7	12.0	0.0	-0.04	0.03	0.51	0.17	1.2
			(5.1)	(0)	(-2.6)	(19.4)	(6.1)	(9.8)	(17.6)

* NOTE: Negative sign indicates computed variable is smaller than measured variable.

Craft total weight was 6040 lbs. in all runs.

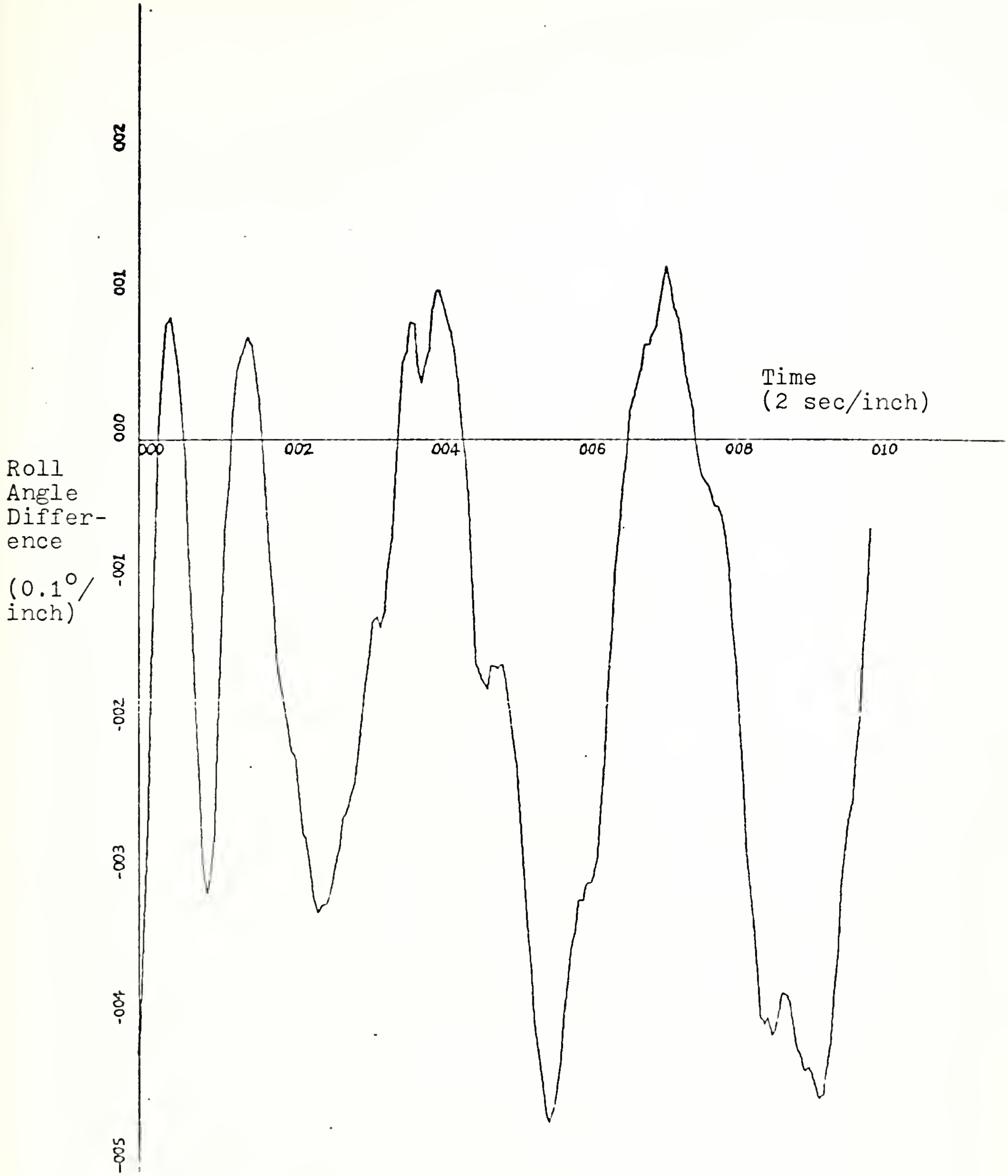


Figure 19. Plot of Roll Angle Difference Vs. Time

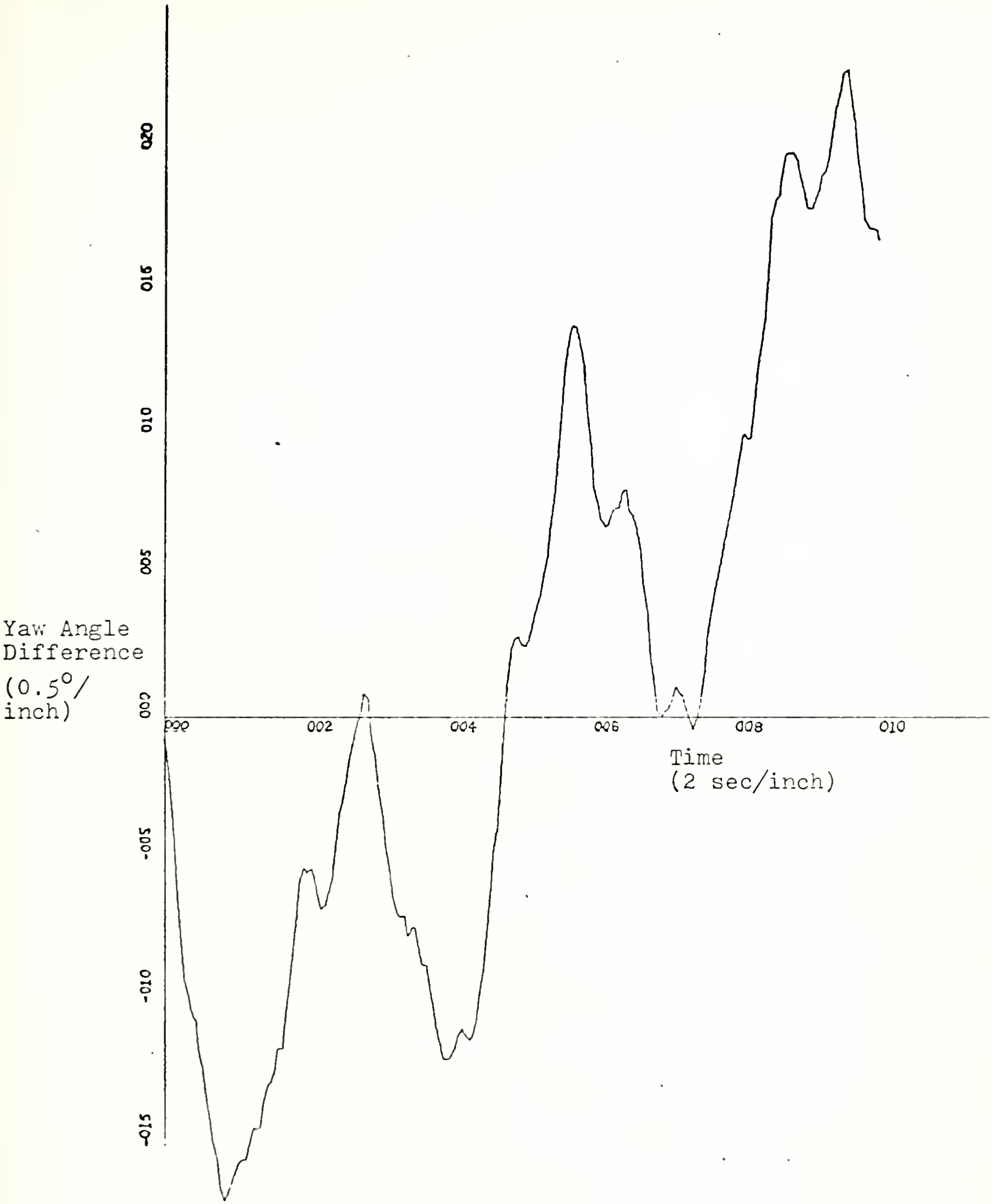


Figure 20. Plot of Yaw Angle Difference Vs. Time

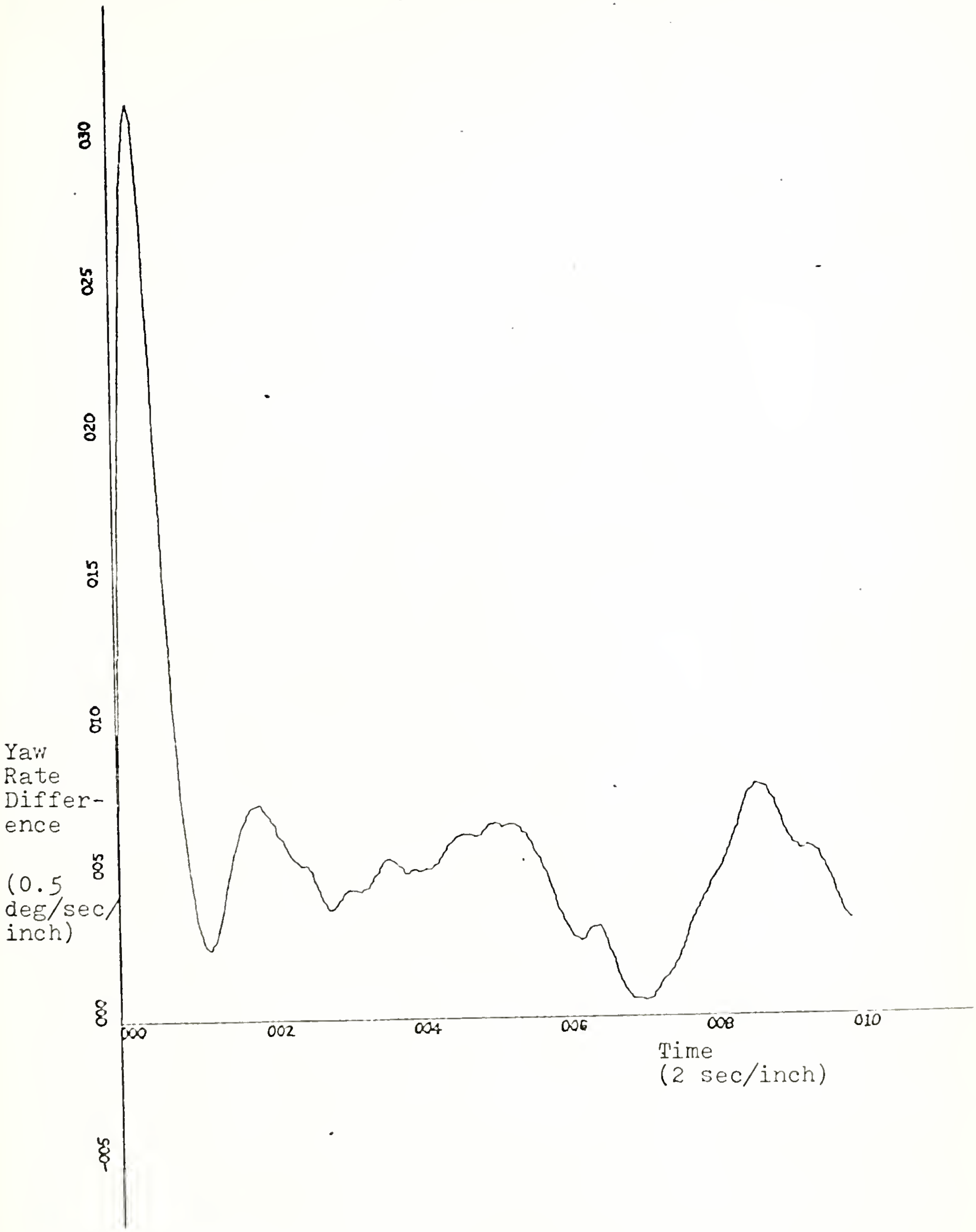


Figure 21. Plot of Yaw Rate Difference Vs. Time

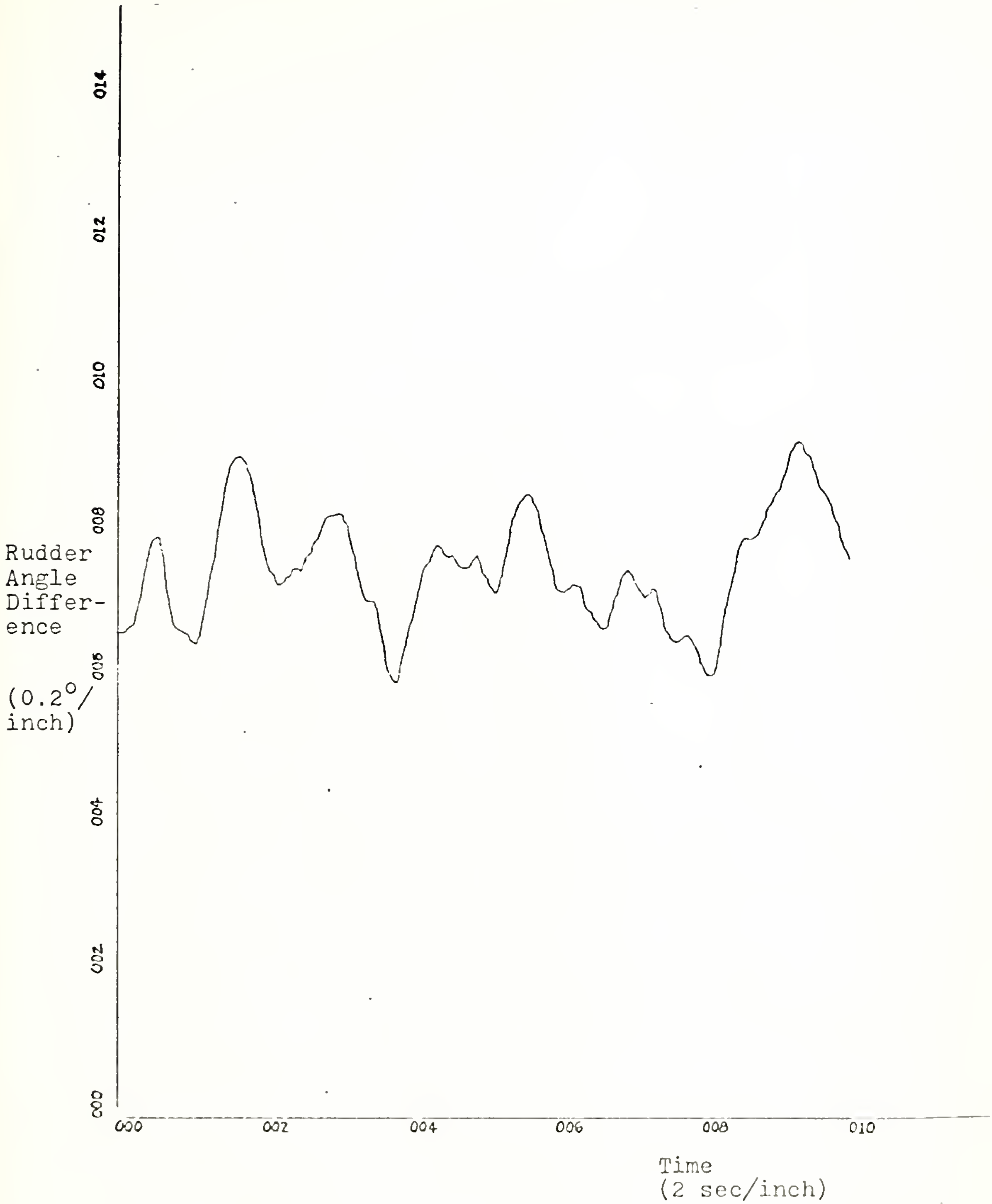


Figure 22. Plot of Rudder Angle Difference Vs. Time

A listing of the FORTRAN program for simulation of the XR-3 is included as Appendix D. A sample input data deck is also included.

VII. CONCLUSIONS AND RECOMMENDATIONS

This study has shown that the nonlinear mathematical model with six degrees of freedom adequately describes the motions of the XR-3 in calm water conditions. The variables used in the validation included thrust, longitudinal velocity, pitch angle, plenum pressure, draft and vertical acceleration in straight runs. For validation of turning dynamics the variables included thrust, longitudinal velocity, pitch angle, roll angle, yaw angle, yaw rate and rudder angle. Differences that were found to exist between the computer model and the testcraft were less than 5% for the majority of the variables compared and those differences that exceeded 5% are considered acceptable, given the range of the measured variables and the accuracy of the sensors aboard the testcraft.

During the course of this study, it became apparent that the computer model exhibits a more oscillatory, or underdamped, response in pitch and roll motions than does the XR-3 testcraft. A concurrent study by LCDR. R.A. Finley, USN upon this aspect of the simulation with attention to the modelling of the bow seal and the lift fan system is reported in Reference 12.

The method of data processing set forth in this study to produce data in adequate form for interface with the mathematical model is recommended for use in future

validation efforts. A considerable amount of time and effort was spent in signal analysis and data processing to prepare the measured data in a form useful for comparison with the mathematical model variables. In order to eliminate the requirement for smoothing the re-scaled data, it is recommended that a study be made using a lower sampling rate for the A-to-D conversion to see if the high frequency noise can be eliminated from the digitally reproduced data. It is also recommended that a vibration study be conducted aboard the XR-3 in an attempt to determine if the source or sources of the high frequency noise is the result of internal ship vibration acting on the measurement transducers, gyros, and accelerometers. The current data acquisition system installed in the testcraft should be adequate for such a study.

A very fertile area for future study would be validation of the computer model in the presence of waves. Instrumentation exists to measure wave conditions at the XR-3 test site during measurement runs and the simulated waves could be input to the mathematical model. Direct comparison of craft behavior and simulation model calculations could then be obtained to produce an evaluation of the accuracy of the simulation in sea states.

APPENDIX A

LISTING OF FORTRAN PROGRAM FOR ANALOG TO DIGITAL CONVERSION OF MEASURMENT DATA

```
      DIMENSION IBUF(4096,2),LOCB(-1'1),BADREC(100)
      INTEGER RECNUM,BADREC
      NAMELIST NREC, NSAMP, NCHAN, ITAPE,NDEL,NT
1     INPUT(101)
      NWORDS=NSAMP*NCHAN
      LOCB(-1)=LOCF(IBUF(1,1))
      IF NREC.GT.1,LOCB(1)=LOCF(IBUF(1,2))
      IF NREC.EQ.1,LOCB(1)=LOCB(-1)+NWORDS
      IND=2
      GO TO 15
2     NB=1
      RECNUM=0
      NEWBUF=LOCB(1)
      IF(IABS(IND).NE.2)GO TO 94
      CALL WRITECLOCK(0)
      CALL ADSTART(NCHAN,LOCB(-1),NEWBUF,NSAMP,RECNUM,11S)
      CALL MTRDY(ITAPE,LOCB(-1),LOCB(1),NWORDS,IND)
4     IF IND.EQ.1, GO TO 4
      GO TO(4,3,91,93)IND
3     IF NT.EQ.0,GO TO 7
      CALL MTRDY1(ITAPE+1,LOCB(-1),LOCB(1),NWORDS,IND)
6     IF IND.EQ.1,GO TO 6
      GO TO(6,7,91,93)IABS(IND)
7     IF(TEST(1).GT.0)GO TO 7
      IND=2
      NBAD=0
      CALL STARTCLOCK
      CALL ENABLE
5     CONTINUE
10    GO TO 5
11    NB=-NB
      NEWBUF=LOCB(NB)
      IF(TEST(1).GT.0.OR.RECNUM.GE.NREC)CALL DISABLE
      GO TO(90,12,91,92)IABS(IND)
12    CONTINUE
      IF(IND.GT.0)CALL MTOUT(NB)
      IF(IND.LT.0)CALL MTOUT1(NB)
      IF(TEST(1).LT.0.AND.RECNUM.LT.NREC)GO TO 5
      CALL STOPCLOCK
      CALL ADSTOP
      CALL PROCESS(IBUF,NSAMP,NCHAN+1,2S)
      OUTPUT( 6 )RECNUM
      IF NBAD.NE.0,WRITE( 6 ,106) (BADREC(I),I=1,NBAD)
106   FORMAT($ BAD RECORDS ARE$,(/,I6))
15    OUTPUT(102)'OPTION=(I1)'
      READ(101,100)NOPT
100   FORMAT(I1)
      GO TO(1,2,30,40,50,60,70)NOPT
30    OUTPUT(102)'EOF ON WHICH TAPE'
      READ(101,100)N
      IF N.EQ.0,GO TO 15
      ENDFILE(N)
      GO TO 15
40    OUTPUT(102)'REWIND WHICH TAPE'
      READ(101,100)N
      IF N.EQ.0,GO TO 15
      REWIND(N)
      GO TO 15
50    OUTPUT(102)'SKIPFILES=(I4)'
      READ(101,101)NF
101   FORMAT(I4)
      OUTPUT(102)'ON WHICH TAPE'
```

```
      READ(101,100)N
      IF N.EQ.0,GO TO 15
      DO 55 I=1,NF
51    CALL BUFFERIN(N,1,IBUF(1,1),1,IN )
52    IF(IN .LT.2)GO TO 52
      IF(IN .NE.3)GO TO 51
55    CONTINUE
      OUTPUT(102)NF
      GO TO 15
60    OUTPUT(102)'NUMWORDS TO LIST=(I4)'
      READ(101,101)NW
      OUTPUT(102)'FROM WHICH TAPE'
      READ(101,100)N
      IF N.EQ.0,GO TO 15
601   IN =1
      CALL BUFFERIN(N     ,1,IBUF(1,1),NWORDS,IN )
66    IF(IN .EQ.1)GO TO 66
62    GO TO(62,63,64,65)IN
63     WRITE(6,102)
102   FORMAT(1H1)
      DO 631 I=1,NW,NCHAN
      WRITE(6,104)(IBUF(J,1),J=I,I+NCHAN-1)
104    FORMAT(12O10)
631   CONTINUE
      IF (SENSE SWITCH 1)601,15
64     OUTPUT(102)'EOF READ'
      GO TO 15
65     OUTPUT(102)'READ ERR'
      GO TO 63
70    OUTPUT(102)'D/A FROM WHICH TAPE'
      READ(101,100)N
      IF N.EQ.0,GO TO 15
      OUTPUT(102)'TYPE * C/R TO CONTINUE'
      INPUT(101)
77    IN =1
      CALL BUFFERIN(N     ,1,IBUF,NWORDS,IN )
76    IF(IN .EQ.1)GO TO 76
71    GO TO(71,72,64,74)IN
72    DO 73 I=1,NWORDS
73    IBUF(I,1)=IBUF(I,1)/2**10
      DO 75 I=1,NWORDS,NCHAN
      DO 750 J=1,NCHAN
750   CALL DAC(J,IBUF(I+J-1,1))
      M=NDEL
      CALL DELAY
75    CONTINUE
      IF(SENSE SWITCH 1)77,15
74    OUTPUT(102)'READ ERROR'
      GO TO 72
90    CALL DISABLE
      CALL ADSTOP
      OUTPUT(102)'RATE ERR',RECNUM
      GO TO 15
92    IF NBAD.LT.100,NBAD=NBAD+1
      BADREC(NBAD)=RECNUM-1
      GO TO 12
91    CALL DISABLE
      CALL ADSTOP
      OUTPUT(102) - MT NOT READY -
      GO TO 15
93    CALL DISABLE
      CALL ADSTOP
      OUTPUT(102) - MT ERROR ON SPACING FROM LOAD POINT-
      GO TO 15
94    OUTPUT(102)'DELAY TIME BETWEEN RECORDS TOO SHORT'
      GO TO 15
      END
```

```
      SUBROUTINE PROCESS(I,N,NC,IR)
      J=N*NC*9
      CALL DELAY
1     IF TEST(1).LT.0,GO TO 1
      IF TEST(2).GT.0,RETURN IR
      RETURN
      END
```

APPENDIX B

LISTING OF FORTRAN PROGRAM TO READ FROM SEVEN TRACK TAPE, CONVERT FROM OCTAL TO HEXADECIMAL, AND STORE RESULT ON DISK

```
//CONVERT EXEC FORTCLG,REGION.GO=100K
//FORT.SYSIN DD *
      DIMENSION IDAT(3072), DAT(3072)
      FACTOR=100./(2**23)
      REWIND 2
      NRECL = 3072
      J=0
   10 READ(2,15,END=50,ERR=60) IDAT
  15  FORMAT (16(192A4))
      J=J+1
      WRITE (6,70) J
   70 FORMAT ('0',10X,'RECORD NO.=',I4)
      CALL FORM(IDAT,NRECL)
      DO 22 I=1,NRECL
   22 DAT(I)=IDAT(I)*FACTOR
      WRITE (6,40) (DAT(I),I=1,180)
  40  FORMAT (1X,6F12.5)
      WRITE (4) DAT
      GO TO 10
   60 WRITE(6,61) J
   61 FORMAT ('0',5X,'READ ERROR RECORD NO.=',I3)
      GO TO 10
   50 WRITE (6,51) J
   51 FORMAT('0',5X,'END OF TAPE, RECORD NO.=',I3)
      ENDFILE 4
      RETURN
      END

//GO.FT02F001 DD UNIT=2400-1,VOL=SER=BOTEY,LABEL=(9,NL,,IN),
// DISP=OLD,DCB=(DEN=1,RECFM=F,BLKSIZE=12288)
//GO.FT04F001 DD UNIT=2314,DSNAME=S0896.TAPESS,DISP=(NEW,KEEP),
// VOL=SER=DUFFY,
// LABEL=RETPD=120,SPACE=(TRK,(2,2),RLSE),
// DCB=(RECFM=VS,BLKSIZE=3504,LRECL=3000)
```

APPENDIX C

LISTING OF FORTRAN PROGRAM TO READ FROM DISK, RESCALE DATA TO MEASURED UNITS, PERFORM SMOOTHING, AND STORE THE RESULT ON DISK

```
// EXEC FORTCLGP,REGION.GO=200K
//FORT.SYSIN DD *
      INTEGER*4 ITB(12)/12*0/
      REAL*4 RTB(28)/28*0.0/
      EQUIVALENCE (TITLE,RTB(5))
      REAL*8 TITLE(12)
      DIMENSION TP(2000),TS(2000),HT(2000),PANG(2000),PB(2000),
     1VK(2000),TTHST(200),PITCH(200),PBUB(200),SPD(200),DRFT(200),
     2TIME(200)
      READ (5,110) NPTS
 110  FORMAT (I4)
      M=500
      N=1
      DO 5 K=1,4
      READ (4) (TP(I),TS(I),HT(I),PANG(I),VK(I),PB(I),I=N,M)
      N=500+N
      M=500+M
  5   CONTINUE
      CALL SMOOTH (TP)
      CALL SMOOTH (TS)
      CALL SMOOTH (HT)
      CALL SMOOTH (PANG)
      CALL SMOOTH (VK)
      CALL SMOOTH (PB)
      DC 10 J=1,200
      K=J
      TTHST(J)=(TP(K)+TS(K))*10.
      PITCH(J)=(PANG(K)-25.)*0.6
      PBUB(J)=PB(K)*1.2
      SPD(J)=VK(K)*0.8
      HT(J)=(100.-HT(K))*0.48
      DRFT(J)=48.0-HT(J)-3.19335*PITCH(J)
      TIME(J)=(J-1)*0.05
 10   CONTINUE
      WRITE(6,165) (I,TTHST(I),DRFT(I),PITCH(I),SPD(I),PBUB(I),I=1,200)
```

```
165   FORMAT(//3X,I4,5F12.4)
      WRITE(8)(TTHST(I),DRFT(I),PITCH(I),SPD(I),PBUB(I),I=1,200)
      READ (5,100) TITLE
100   FORMAT (6A8)
      CALL DRAWP(NPTS,TIME,TTHST,ITB,RTB)
      READ (5,100) TITLE
      CALL DRAWP(NPTS,TIME,PITCH,ITB,RTB)
      READ (5,100) TITLE
      CALL DRAWP(NPTS,TIME,PBUB  ,ITB,RTB)
      READ (5,100) TITLE
      CALL DRAWP(NPTS,TIME,SPD   ,ITB,RTB)
      READ (5,100) TITLE
      CALL DRAWP(NPTS,TIME,DRFT  ,ITB,RTB)
      STOP
      END


      SUBROUTINE SMOOTH (VBLE)
      DIMENSION VBLE(2000)
      SUM=0.0
      N=1
      M=10
      DO 210 I=1,200
      DO 205 K=N,M
      SUM=SUM+VBLE(K)
205   CONTINUE
      VBLE(I)=0.1*SUM
      N=N+5
      M=M+5
      SUM=0.0
210   CONTINUE
      RETURN
      END

//GO.FT04F001 DD DISP=SHR,DSN=S0896.TAPESS,UNIT=2314,VOL=SER=DUFFY,
// LABEL=(,,,IN),DCB=(RECFM=VS,BLKSIZE=3504,LRECL=3000)
//FT03F001 DD DSN=S0896.DATA10,UNIT=2314,VOL=SER=MARY,DISP=(NEW,KEEP),
// LABEL=RETPD=180,DCB=(RECFM=VS,LRECL=4004,BLKSIZE=7290),
//  SPACE=(TRK,(2,1))
```

APPENDIX D

LISTING OF FORTRAN XR-3 SIMULATION PROGRAM

```
      INTEGER ON

      COMMON /AIR/ PINF,RHOINF,GAM
      COMMON /BMCO / IMM,IMNX,IMNY,IBMFIL,BTIME,IMT,XMI(10),YMI(7),IX,IY
      COMMON /CONST/ PI,RAD,UO
      COMMON/ENGINE/NPS,NPP,THSTS(25),THSTP(25),XP,YP,ZP,STHS,STHP,
     ATIP(25),TIS(25)
      COMMON /EQNCO/ NEQS,TOL(20),JQQ
      COMMON /FPROP/ FXP,FYP,FZP,FKP,FMP,FNP
      COMMON / FROUDE / FN,FNCKIT
      COMMON /PRIME/ STIME,FTIME,DELT,DELPNT,TPRINT
      COMMON /PRTINT/ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,
     -IRUD,IPRUP,IAERUD,IRHS
      COMMON /ROLL/ PHIMAX,TROLL
      COMMON/RUDDR/ NPR,DELRUD(25),XR,YR,ZR,IRDS,TL,RSPAN,RAREA,RASPR,
     ARCLB,RTC,RUDANG,TIR(25)
      COMMON / VALOLD / YOLD(20)
      COMMON /VARBLE/ VAL(40)
      COMMON /WAVE/ ETA(4,11),AW(10),OMEGA(10),DVCLW,NWAVE,BETA,
     1               FXWAV,FYWAV,FZWAV,FKWAV,FMWAV,FNWAV
     2               ,ZBAR,PHIBAR,THEBAR,TC,COSBET,SINBET,PBBAR

      EQUIVALENCE             (VAL(2),U),(VAL(3),V),(VAL(4),W),
     1(VAL(5),P),(VAL(6),Q),(VAL(7),R),(VAL(8),PHI),(VAL(9),THETA),
     2(VAL(10),Z),(VAL(11),BMASS),(VAL(21),X),(VAL(22),Y),(VAL(23),PSI),
     3(VAL(24),PB)
      DIMENSION DUMMY(20)

      TC=1.0
      ON=1
      PI=4.*ATAN(1.)
      RAD=180./PI
      WRITE(6,100)
100   FORMAT(1H1//35X,22H LISTING OF INPUT DECK  //)
99    READ(44,101,END=104) DUMMY
101   FORMAT(20A4)
105   WRITE(6,102)DUMMY
      WRITE(5,101) DUMMY
102   FORMAT(5X,20A4)
      GO TO 99
104   REWIND 5
```

```
11    CALL INCON(TIME)
      IF (IMM.EQ.3)  GO TO 605

      DO 10 J=1,20
 10   YOLD(J)=VAL(J+1)
      GO TO 2

  1   CONTINUE
      TOLD=TIME
      PBBAR=PBBAR*(1.-DELT/TC)+DELT*(PB-PINF)/TC
      IF ( NWAVE.LE.0 )  GO TO 13
      ZBAR=(1.-DELT/TC)*ZBAR+DELT*Z/TC
      PHIBAR=(1.-DELT/TC)*PHIBAR+DELT*PHI/TC
      THEBAR=(I.-DELT/TC)*THEBAR+DELT*THETA/TC
      CALL WAVES(TIME)

  13 CALL SIDEWL
      CALL PROP
      CALL RUDDER
      CALL AEROD
      CALL INTGRL(TIME)
      IF(TIME.GT.FTIME) GO TO 12
      IF ( FN.GT.FNCRIT) GO TO 14
      PRINT 505
      GO TO 12

  14 DELOLD=TIME-TOLD
      PSI=PSI+DELOLD*R
      X=X+DELOLD*(U*COS(PSI)-V*SIN(PSI))
      Y=Y+DELOLD*(U*SIN(PSI)+V*COS(PSI))
 15   IF (ABS(TIME-TPRINT) .LT. 1.E-6) GO TO 2
      GO TO I

  2   CONTINUE
      IF(ITRAJ .EQ. 0) GOTO 16
      DPHI=PHI*RAD
      DPSI=PSI*RAD
      DTHETA=THETA*RAD
      DP=P*RAD
      DQ=Q*RAD
      DR=R*RAD
      VEL=0.5925*U
      WRITE (6,500) TIME,VEL,V,W,DP,DQ,DR,Z,DPHI,DTHETA,X,Y,DPSI
      BETS=(-V/U)*RAD
      DELRS=RUDANG*RAD
      WRITE(6,501) BETS,DELRS,FXP
16    CONTINUE
      IMMTAG = (IMM+1)/2
```

```
      IF ( IMMTAG .EQ. 1  .AND. TIME.GE.BTIME-1.E-8 )  IMT = 1
      TPRINT=TPRINT+DELPNT
      ON=1
      GO TO 1
 12   CALL COLFIL
      IF (IMM.LT.1)  GO TO 11
      IF (IMM.NE.1)  GO TO 605
      END FILE IBMFIL
      GO TO 11
 605  CALL SAM
      GO TO 11
 500  FORMAT(//10X,13HTIME (SEC) =  F6.2//10X,33HTRANSLATIONAL VELS (KTS
     1)/(FT/SEC) /10X,2HU= F6.2,5X,2HV= F6.2,5X,2HW= F6.3//10X,31HROTATI
     2ONAL VELOCITIES (DEG/SEC) /10X,2HP= F6.2,5X,2HQ= F6.2,5X,2HR= F6.2
     3//10X,30HDISPLACEMENTS (FT AND DEGREES) /10X,2HZ= F7.3,5X,4HPHI=
     4F6.2,3X,6HTHETA= F6.2//10X,27HTRAJECTORY (FT AND DEGREES) /10X,
     52HX= F8.2,4X,2HY= F8.2,4X,4HPSI= F8.2)
 501  FORMAT(1H0,9X,23HSIDESLIP ANGLE (DEG) =   F8.2,10X,21HRUDDER ANGLE
     1 (DEG) =  F8.3,10X,15HTHRUST (LBS) =   F12.1)
 505  FORMAT(///25X,28HCRAFT SPEED BELOW HUMP SPEED )
      END
      BLOCK DATA
      COMMON /AIR/ Z1(3)
      COMMON /BMCO/ Z2(25)
      COMMON /COLUMN / Z3(2)
      COMMON /CONST/ Z4(3)
      COMMON /CNTRL/ Z5(10)
      COMMON /ENGINE/ Z6(107)
      COMMON /EQNCO/ Z7(22)
      COMMON /FAERO/ Z8(6)
      COMMON /FAIR/ Z9(2)
      COMMON /FANMAP/ Z10(262)
      COMMON /FORBS/ Z11(7)
      COMMON /FORSS/Z12(8)
      COMMON /FPROP/ Z13(6)
      COMMON /FROUDE/ Z14(2)
      COMMON /FRUD/ Z15(6)
      COMMON /GBOW/ Z16(1)
      COMMON /GEOM/ Z17(138)
      COMMON /GEOMBS/ Z18(62)
      COMMON /GEOMSS/ Z19(62)
      COMMON /GEOMSW/ Z20(11)
      COMMON /KSWTCH/ Z21(1)
      COMMON /LEAKER/ Z22(4)
      COMMON /MASSES/ Z23(817)
      COMMON /MATRIX/ Z24(36)
      COMMON /MSIDW/ Z25(55)
```

```
      COMMON /MWAVE/ Z26(12)
      COMMON /OPTION/ Z27(4)
      COMMON /PLENUM/ Z28(4)
      COMMON /PLVCQQ/ Z28A(4)
      COMMON /PRIME/ Z29(5)
      COMMON /PRTINT/ Z30(12)
      COMMON /PWAVE/ Z31(2)
      COMMON /RISER/ Z32(1)
      COMMON /ROLL/ Z33(2)
      COMMON /RUDDR/ Z34(62)
      COMMON /SIDE/ Z35(22)
      COMMON /SOFTBS/ Z36(19)
      COMMON /SOFTSS/ Z37(19)
      COMMON /STABLE/ Z38(5)
      COMMON /STSLR/ Z39(2)
      COMMON /VALOLD/ Z40(20)
      COMMON /VARBLE/ Z41(40)
      COMMON /WAVE/ Z42(80)
      COMMON /WAVEF / Z43(40)
      COMMON /SLOPE/Z44(4)
      DATA Z1/3*0.0/
      DATA Z2/25*0.0/
      DATA Z3/2*0.0/
      DATA Z4/3*0.0/
      DATA Z5/10*0.0/
      DATA Z6/107*0.0/
      DATA Z7/21*0.0/
      DATA Z8/6*0.0/
      DATA Z9/2*0.0/
      DATA Z10/262*0.0/
      DATA Z11/7*0.0/
      DATA Z12/8*0.0/
      DATA Z13/6*0.0/
      DATA Z14/2*0.0/
      DATA Z15/6*0.0/
      DATA Z16/0.0/
      DATA Z17/138*0.0/
      DATA Z18/62*0.0/
      DATA Z19/62*0.0/
      DATA Z20/11*0.0/
      DATA Z21/0.0/
      DATA Z22/4*0.0/
      DATA Z23/817*0.0/
      DATA Z24/36*0.0/
      DATA Z25/55*0.0/
      DATA Z26/12*0.0/
      DATA Z27/4*0.0/
      DATA Z28/4*0.0/
```

```
      DATA Z28A/4*0.0/
      DATA Z29/5*0.0/
      DATA Z30/12*0.0/
      DATA Z31/2*0.0/
      DATA Z32/0.0/
      DATA Z33/2*0.0/
      DATA Z34/62*0.0/
      DATA Z35/22*0.0/
      DATA Z36/19*0.0/
      DATA Z37/19*0.0/
      DATA Z38/5*0.0/
      DATA Z39/2*0.0/
      DATA Z40/20*0.0/
      DATA Z41/40*0.0/
      DATA Z42/80*0.0/
      DATA Z43/40*0.0/
      DATA Z44/4*0.0/
      END


      SUBROUTINE AEROD
      INTEGER ON
      COMMON /FAERO/ FX,FY,FZ,FK,FM,FN
      COMMON /FAIR/ RHOA,XLAERO
      COMMON /PRTINT/ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,
     -IRUD,IPROP,IAEROD,IRHS
      COMMON /VARBLE/ VAL(40)

      EQUIVALENCE              (VAL(2),U),(VAL(3),V),(VAL(4),W),
     1(VAL(5),P),(VAL(6),Q),(VAL(7),R),(VAL(8),PHI),(VAL(9),THETA),
     2(VAL(10),Z),(VAL(11),BMASS),(VAL(21),X),(VAL(22),Y),(VAL(23),PSI),
     3(VAL(24),PB)
      QA=RHOA*U*U
      QAL=QA*XLAERO
      BETA=-V/U
      BETASQ=BETA*BETA
      FX=-(0.90*BETASQ+0.13)*QA
      FY=(0.0*BETASQ+0.53*BETA)*QA
      FZ=-(2.06*BETASQ+0.39)*QA
      FK=-(0.5*BETASQ+0.0*BETA)*QAL
      FM=(0.29*BETASQ+0.12)*QAL
      FN=(0.0*BETASQ+0.076*BETA)*QAL

      IF (IAERUD.NE.ON) RETURN
                 WRITE(6,100) FX,FY,FZ,FK,FM,FN
  100 FORMAT(/10X,23HAEROD FX,FY,FZ,FK,FM,FN/6E15.4)
      RETURN
      END
```

```
      SUBROUTINE BOWSL                                                  BWSL0010
      INTEGER ON                                                        BWSL0020
      COMMON /AIR/ PINF,RHOINF,GAM                                      BWSL0030
      COMMON /CONST/ PI,RAD,U0                                          BWSL0040
      COMMON / FORBS/ FX,FY,FZ,FK,FM,FN,QL                              BWSL0050
      COMMON /GEOM/    WIDTH,XL,XX(4,11),YY(4,11),NSTA(4),AB,VOLNOM     BWSL0060
     1,DELS(4,10),XCP,ZCP                                               BWSL0070
      COMMON /GEOMBS/DETABX(11),DETABT(11),ARM1B(10),ARM2B(10)          BWSL0080
     1 ,DFBS(10),TSKIB(10)                                              BWSL0090
      COMMCN/LEAKER/ALEAK,BLEAK,CFSS,CFBS                               BWSL0100
      COMMON /MASSES/ AM,AIXX,AIYY,AIZZ,AIXZ,AIMAX,G,WEIGHT,RHO,NMASS,  BWSL0110
     -                AMI(201),XI(201),YI(201),ZI(201),XS,ZS,HRHO       BWSL0120
      COMMON /PRTINT/ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,  BWSL0130
     -IRUD,IPROP,IAEROD,IRHS                                            BWSL0140
      COMMON /SLOPE/WATSLP,XPWV,XLXPWV,XPWVXS                           BWSL0150
      COMMON/SUFTBS/XBF,PBS,SINBS,COSBS,XBS,ZBS,DELYBS,DPBS,ELMAXB,YAVG BWSL0160
     1B(10)                                                             BWSL0170
      COMMON /VARBLE/ VAL(40)                                           BWSL0180
      COMMCN /WAVE/ ETA(4,11),AW(10),OMEGA(10),DVOLW,NWAVE,BETA,        BWSL0190
     1               FXWAV,FYWAV,FZWAV,FKWAV,FMWAV,FNWAV                BWSL0200
     2               ,ZBAR,PHIBAR,THEBAR,TC,COSBET,SINBET,PBBAR         BWSL0210
      DIMENSION GAP(11),ELSKI(11)                                       BWSL0220
      DIMENSION WTAB(6),ZTAB(6)                                         BWSL0230
      DATA NPTS,IBS/6,0/                                                BWSL0240
      DATA WTAB/0.0,3.75,5.42,6.67,7.5,8.42/                            BWSL0250
      DATA ZTAB/3.75,4.00,4.42,4.83,5.25,5.67/                          BWSL0260
      DATA ENU,UWSKI,CLSKI/1.28E-05,0.0,1.5708/                         BWSL0270
      EQUIVALENCE (VAL(1),TIME),(VAL(2),U),(VAL(3),V),(VAL(4),W),       BWSL0280
     1(VAL(5),P),(VAL(6),Q),(VAL(7),R),(VAL(8),PHI),(VAL(9),THETA),     BWSL0290
     2(VAL(10),Z),(VAL(11),BMASS),(VAL(21),X),(VAL(22),Y),(VAL(23),PSI),BWSL0300
     3(VAL(24),PB)                                                      BWSL0310
      DO 5 J=1,11                                                       BWSL0320
      GAP(J)=0.0                                                        BWSL0330
      ELSKI(J)=0.0                                                      BWSL0340
    5 CONTINUE                                                          BWSL0350
      ALBS=0.0                                                          BWSL0360
      FX=0.0                                                            BWSL0370
      FZ=0.0                                                            BWSL0380
      FK=0.0                                                            BWSL0390
      FM=0.0                                                            BWSL0400
      FN=0.0                                                            BWSL0410
      DELPBG=PBS-PB                                                     BWSL0420
      IF(DELPBG.LT.0.0) DELPBG=0.0                                      BWSL0430
      PBAR=PB-PINF                                                      BWSL0440
      DELP=PBAR                                                         BWSL0450
      IF(DELP.LT.0.0)DELP=0.0                                           BWSL0460
```

```
      SINDIF=SINBS-COSBS*THETA                                          BWSL0470
      COSDIF=COSBS+SINBS*THETA                                          BWSL0480
      X1=XBS+ZBS*THETA-XBF*SINDIF                                       BWSL0490
      Z1=-Z-ZBS+XBS*THETA-XBF*COSDIF                                    BWSL0500
      N=NSTA(3)                                                         BWSL0510
      DO 10 K=1,N                                                       BWSL0520
      ELSKI(K)=(ETA(3,K)-DETABX(K)*(XX(3,K)-X1)-Z1)+YY(3,K)*PHI         BWSL0530
     1+XLXPWV*WATSLP                                                    BWSL0540
      GAP(K)=-ELSKI(K)                                                  BWSL0550
      IF(GAP(K).LT.0.0) GAP(K)=0.0                                      BWSL0560
   10 CONTINUE                                                          BWSL0570
      N=NSTA(3)-1                                                       BWSL0580
      DO 20 J=1,N                                                       BWSL0590
      ELSKIA=(ELSKI(J+1)+ELSKI(J))/2.0                                  BWSL0600
      IF(ELSKIA.LE.0.0) GO TO 15                                        BWSL0610
      IF(ELSKIA.GT.ELMAXB) ELSKIA=ELMAXB                                BWSL0620
      ARM1B(J)=XX(3,J)+ELSKIA/2.0                                       BWSL0630
      ARM2B(J)=ZS-ELSKIA                                                BWSL0640
      DFBS(J)=-DELP*ELSKIA*DELYBS                                       BWSL0650
      ARG=0.5*RHO*U*U*ELSKIA*DELYBS                                     BWSL0660
      RESKI=U*ELSKIA/ENU                                                BWSL0670
      CDTSKI=0.427/(ALOG10(RESKI)-0.407)**2.64                          BWSL0680
      TSKIB(J)=-ARG*CDTSKI                                              BWSL0690
      GO TO 18                                                          BWSL0700
   15 DFBS(J)=0.0                                                       BWSL0710
      TSKIB(J)=0.0                                                      BWSL0720
   18 CONTINUE                                                          BWSL0730
      FX=FX+TSKIB(J)                                                    BWSL0740
      FZ=FZ+DFBS(J)                                                     BWSL0750
      FK=FK+DFBS(J)*YAVGB(J)                                            BWSL0760
      FM=FM-DFBS(J)*ARM1B(J)+TSKIB(J)*ARM2B(J)                          BWSL0770
      FN=FN-TSKIB(J)*YAVGB(J)                                           BWSL0780
      ALBS=ALBS+(GAP(J)+GAP(J+1))*DELYBS/2.0                            BWSL0790
   20 CONTINUE                                                          BWSL0800
      ALBS=ALBS+BLEAK                                                   BWSL0810
      QL=CFBS*ALBS*SQRT(2.0*ABS(PBAR)/RHOINF)*SIGN(1.0,PBAR)            BWSL0820
      IF(IBOWSL.NE.ON) RETURN                                           BWSL0830
      WRITE (6,100) GAP,ELSKI,FX,FY,FZ,FK,FM,FN                         BWSL0840
  100 FORMAT(//10X,8HBOW SEAL /26H GAP (FT.) (PORT TO STBD.) /11F10.5   BWSL0850
     1/28H ELSKI (FT.) (PORT TO STBD.) /11F10.5 /10X,23HBOWSL FX,FY,FZ,FBWSL0860
     2K,FM,FN /6E15.4)                                                  BWSL0870
      RETURN                                                            BWSL0880
      END                                                               BWSL0890

      SUBROUTINE COLFIL                                                 CFL 0020
      COMMON/AXIS/NXYS(26)                                              CFL 0070
      COMMON/COLUMN/IVERT,ILATRL                                        CFL 0080
```

```
      COMMON /CURVE/NCURV(10)
      COMMON/EQNCO/ NEQS,TOL(20),JQQ
      COMMON/GRAF/NGRAF,NDRW
      COMMON /HEADG/TICRD(6)
      COMMON/STEP/ STEP2
      COMMON /SUM/ ISUM1(8),ISUM2(8)
      REAL*8 TICRD
      REAL LABEL
      REAL LAB(4)/'    ','   1','   2','   3'/
      REAL*8 NAMES(52)/'TIME    ','        ','WAVE HEI','GHT     ','Z DI
     1SPLA','CEMENT  ','PITCH AN','GLE     ','PLENUM P','RESSURE ','BOW 
     2ACCE' ,'LERATION','C.G.ACCE','LERATION','FAN POWE','R       ','ROL
     3L ANG','LE      ','YAW ANGL','E       ','LAT ACCE','LERATION','SPE
     4ED TH','RU WATER','TURN RAD','IUS     ','TURN RAT','E       ','X D
     5ISPLA','CEMENT  ','Y DISPLA','CEMENT  ','AIR FLOW',' IN     ','AIR
     6 FLOW',' OUT    ','NET FORC','E X DIR ','WAVE FOR','CE X DIR','NET
     7 FORC','E Y DIR ','NET FORC','E Z DIR ','NET TORQ','UE XAXIS','NET
     8 TORQ','UE YAXIS','NET TORQ','UE ZAXIS','RUDDER A','NGLE    '/
      REAL*8 TITLE(12)
      REAL*8 LINE2(2)/'PLOT IS ',' VERSUS '/
      REAL*8 NAMEX(2),NAMEY(2),INAME(16)
      EQUIVALENCE (TITLE(1),TICRD(1)),(TITLE(2),TICRD(2)),(TITLE(3),TI
     1CRD(3)),(TITLE(4),TICRD(4)),(TITLE(5),TICRD(5)),(TITLE(6),TICRD(
     26))
      DIMENSION PVQQ(26),XOUT(900),YOUT(900),AFILE(8)
      EQUIVALENCE (PVQQ(1),TIME),(PVQQ(2),ETA),(PVQQ(3),Z),(PVQQ(4),THET
     1A),(PVQQ(5),PB),(PVQQ(6),BOWACC),(PVQQ(7),ACC),(PVQQ(8),FANPWR),(P
     2VQQ(9),PHI),(PVQQ(10),BETAS),(PVQQ(11),ACCLAT),(PVQQ(12),U),(PVQQ(
     313),TRADUS),(PVQQ(14),R),(PVQQ(15),X),(PVQQ(16),Y),(PVQQ(17),QIN),
     4(PVQQ(18),QOUT),(PVQQ(19),GFXXX),(PVQQ(20),FXPWAV),(PVQQ(21),GF2),
     5(PVQQ(22),GF3),(PVQQ(23),GF4),(PVQQ(24),GF5),(PVQQ(25),GF6),
     6(PVQQ(26),DELRS)
      IF(JQQ.NE.2) GO TO 1
      WRITE(6,777) STEP2
  777 FORMAT('0',4X,'THIS RUN USED VARIABLE STEP SIZE',/,'0',4X,'THE MIN
     1IMUM STEPSIZE RECORDED DURING THE RUN WAS',2X,E30.5)
    1 ENDFILE 1
      REWIND 1
      TITLE(7)=LINE2(1)
      TITLE(10)=LINE2(2)
      IF(NGRAF.EQ.0) GO TO 11
      J=1
      NCF=NGRAF
      INDEX=NGRAF*2
      DO 19 I=1,INDEX,2
      INDX=NXYS(I)
      INDY=NXYS(I+1)
      IQ=0
```

```
    7 READ(1,END=8)TIME,ETA,Z,THETA,PB,BOWACC,ACC,FANPWR,PHI,BETAS,ACCLACFL 0570
     1T,U,TRADUS,R,X,Y,QIN,QOUT,GFXXX,FXPWAV,GF2,GF3,GF4,GF5,GF6,DELRS  CFL 0580
      IF(IQ.GE.900) GO TO 8                                             CFL 0590
      IQ=IQ+1                                                           CFL 0600
      XOUT(IQ)=PVQQ(INDX)                                               CFL 0610
      YOUT(IQ)=PVQQ(INDY)                                               CFL 0620
      GO TO 7                                                           CFL 0630
    8 REWIND 1                                                          CFL 0640
      INX=INDX*2                                                        CFL 0650
      INY=INDY*2                                                        CFL 0660
      NAMEX(1)=NAMES(INX-1)                                             CFL 0670
      NAMEX(2)=NAMES(INX)                                               CFL 0680
      NAMEY(1)=NAMES(INY-1)                                             CFL 0690
      NAMEY(2)=NAMES(INY)                                               CFL 0700
      IF(NORW.EQ.1) GO TO 20                                            CFL 0710
      CALL PLOTP(XOUT,YOUT,-IQ,0)                                       CFL 0720
      WRITE(6,301) NAMEX,NAMEY                                          CFL 0730
  301 FORMAT(' ',20X,2A8,' IS THE INDEPENDENT VARIABLE AND ',2A8,' THE DCFL 0740
     1EPENDENT VARIABLE')                                               CFL 0750
      GO TO 19                                                          CFL 0760
   20 TITLE(8)=NAMEY(1)                                                 CFL 0770
      TITLE(9)=NAMEY(2)                                                 CFL 0780
      TITLE(11)=NAMEX(1)                                                CFL 0790
      TITLE(12)=NAMEX(2)                                                CFL 0800
      NCUR=NCURV(J)                                                     CFL 0810
      IF(NCUR.NE.0) GO TO 25                                            CFL 0820
      LABEL=LAB(1)                                                      CFL 0830
      CALL DRAW(IQ,XOUT,YOUT,NCUR,0,LABEL,TITLE,0,0,0,0,0,0,8,8,0,LAST) CFL 0840
      J=J+1                                                             CFL 0850
      GO TO 19                                                          CFL 0860
   25 WRITE(2)IQ,NCUR,NGF,LABEL,(TITLE(K),K=1,12),(XOUT(L),YOUT(L),L=1,ICFL 0870
     1Q)                                                                CFL 0880
      NGF=NGF-1                                                         CFL 0890
   19 CONTINUE                                                          CFL 0900
      IF((NGF.EQ.0).AND.(NCUR.EQ.3)) GO TO 26                           CFL 0910
   11 IF(IVERT.NE.1) GO TO 14                                           CFL 0920
      WRITE(6,100)                                                      CFL 0930
  100 FORMAT('0',50X,'***SUMMARY ONE***',/,'0')                         CFL 0940
      K=0                                                               CFL 0950
      DO 31 I=1,8                                                       CFL 0960
      IF(ISUM1(I).NE.0) K=K+1                                           CFL 0970
   31 CONTINUE                                                          CFL 0980
      NUM1=K                                                            CFL 0990
      IF(K.EQ.0) GO TO 14                                               CFL 1000
      N=K*2                                                             CFL 1010
      J=1                                                               CFL 1020
      DO 33 I=1,NUM1                                                    CFL 1030
      IDEX=ISUM1(I)*2                                                   CFL 1040
```

```
      INAME(J)=NAMES(IDEX-1)                                            CFL 1050
      INAME(J+1)=NAMES(IDEX)                                            CFL 1060
      J=J+2                                                             CFL 1070
   33 CONTINUE                                                          CFL 1080
       WRITE(6,300)(INAME(I),I=1,N)                                     CFL 1090
  300 FORMAT('0',16A8)                                                  CFL 1100
      GO TO 4                                                           CFL 1110
   26 ENDFILE 2                                                         CFL 1120
      REWIND 2                                                          CFL 1130
      NGF=NGRAF                                                         CFL 1140
      J=1                                                               CFL 1150
   27 READ(2,END=38)IQ,NCUR,NGRAF,LABEL,(TITLE(K),K=1,12),(XOUT(L),YOUT(CFL 1160
     1L),L=1,IQ)                                                        CFL 1170
      IF((NGF.EQ.NGRAF).AND.(NCUR.EQ.J)) GO TO 29                       CFL 1180
      GO TO 27                                                          CFL 1190
   29 LABEL=LAB(J+1)                                                    CFL 1200
      CALL DRAW(IQ,XOUT,YOUT,NCUR,0,LABEL,TITLE,0,0,0,0,0,0,8,8,0,LAST) CFL 1210
      REWIND 2                                                          CFL 1220
      J=J+1                                                             CFL 1230
      IF(J.EQ.4) GO TO 38                                               CFL 1240
      GO TO 27                                                          CFL 1250
   38 NGF=NGF-1                                                         CFL 1260
      IF(NGF.EQ.0) GO TO 11                                             CFL 1270
      J=1                                                               CFL 1280
      GO TO 27                                                          CFL 1290
4     READ(1,END=13)(PVQQ(I),I=1,26)                                    CFL 1300
      DO 35 I=1,NUM1                                                    CFL 1310
      J=ISUM1(I)                                                        CFL 1320
   35 AFILE(I)=PVQQ(J)                                                  CFL 1330
      WRITE(6,400)(AFILE(I),I=1,NUM1)                                   CFL 1340
  400 FORMAT('0',8(3X,F10.2,3X))                                        CFL 1350
      GO TO 4                                                           CFL 1360
   13 REWIND 1                                                          CFL 1370
   14 IF(ILATRL.NE.1) GO TO 17                                          CFL 1380
      WRITE(6,200)                                                      CFL 1390
  200 FORMAT('0',50X,'***SUMMARY TWO***',/,'0')                         CFL 1400
      K=0                                                               CFL 1410
      DO 32 I=1,8                                                       CFL 1420
      IF(ISUM2(I).NE.0) K=K+1                                           CFL 1430
   32 CONTINUE                                                          CFL 1440
      NUM2=K                                                            CFL 1450
      IF(K.EQ.0) GO TO 16                                               CFL 1460
      N=K*2                                                             CFL 1470
      J=1                                                               CFL 1480
      DO 34 I=1,NUM2                                                    CFL 1490
      IDEX=ISUM2(I)*2                                                   CFL 1500
      INAME(J)=NAMES(IDEX-1)                                            CFL 1510
      INAME(J+1)=NAMES(IDEX)                                            CFL 1520
```

```
      J=J+2
   34 CONTINUE
       WRITE(6,300)(INAME(I),I=1,N)
   23 READ(1,END=16)(PVQQ(I),I=1,26)
      DO 36 I=1,NUM2
      J=ISUM2(I)
   36 AFILE(I)=PVQQ(J)
      WRITE(6,400)(AFILE(I),I=1,NUM2)
      GO TO 23
   16 REWIND 1
   17 RETURN
      END
C
      SUBROUTINE DMINV (A,N,D)
      DIMENSION A(6,6),PIVOT(6)
      DIMENSION IPVOT(6),INDEX(6,2)
      EQUIVALENCE (IROW,JROW),(ICOL,JCOL)
      D=1.0
      DO 17 J=1,N
      IPVOT(J)=0
   17 CONTINUE
      DO 135 I=1,N
      T=0.0
        DO 9 J=1,N
      IF(IPVOT(J)-1) 13,9,13
   13 DO 23 K=1,N
      IF(IPVOT(K)-1) 43,23,81
   43 IF (ABS(T)-ABS(A(J,K))) 83,23,23
   83 IROW=J
      ICOL=K
      T=A(J,K)
   23 CONTINUE
    9 CONTINUE
      IPVOT(ICOL)=IPVOT(ICOL)+1
      IF(IROW-ICOL) 73,109,73
   73 D=-D
      DO 12 L=1,N
      T=A(IROW,L)
      A(IROW,L)=A(ICOL,L)
      A(ICOL,L)=T
   12 CONTINUE
  109 INDEX(I,1)=IROW
      INDEX(I,2)=ICOL
      PIVOT(I)=A(ICOL,ICOL)
      D=D*PIVOT(I)
      A(ICOL,ICOL)=1.0
```

```
      DO 205 L=1,N                                                      DMV 0350
      A(ICOL,L)=A(ICOL,L)/PIVOT(I)                                      DMV 0360
  205 CONTINUE                                                          DMV 0370
      DO 134 LI=1,N                                                     DMV 0380
      IF(LI-ICOL) 21,134,21                                             DMV 0390
   21 T=A(LI,ICOL)                                                      DMV 0400
      A(LI,ICOL)=0.0                                                    DMV 0410
      DO 89 L=1,N                                                       DMV 0420
      A(LI,L)=A(LI,L)-A(ICOL,L)*T                                       DMV 0430
   89 CONTINUE                                                          DMV 0440
  134 CONTINUE                                                          DMV 0450
  135 CONTINUE                                                          DMV 0460
      DO 3 I=1,N                                                        DMV 0470
      L=N-I+1                                                           DMV 0480
      IF(INDEX(L,1)-INDEX(L,2)) 19,3,19                                 DMV 0490
   19 JROW=INDEX(L,1)                                                   DMV 0500
      JCOL=INDEX(L,2)                                                   DMV 0510
      DO 549 K=1,N                                                      DMV 0520
      T=A(K,JROW)                                                       DMV 0530
      A(K,JROW)=A(K,JCOL)                                               DMV 0540
      A(K,JCOL)=T                                                       DMV 0550
  549 CONTINUE                                                          DMV 0560
    3 CONTINUE                                                          DMV 0570
   81 RETURN                                                            DMV 0580
      END                                                               DMV 0590

C                                                                       FAN 0010
      SUBROUTINE FAN                                                    FAN 0020
      INTEGER ON                                                        FAN 0030
      COMMON /AIR/ PINF,RHOINF,GAM                                      FAN 0040
      COMMON /FANMAP/QIN,QBFAN(25),QMFAN(25),QSFAN(25),ENBFAN,ENMFAN,   FAN 0050
     1 ENSFAN,BRPM,EMRPM,SRPM,NPTSB,NPTSM,NPTSS                         FAN 0060
     2 ,PBFAN(25),PMFAN(25),PSFAN(25),TMEB(25),DELB(25),NB,TMES(25),    FAN 0070
     3DELS(25),NS                                                       FAN 0080
      COMMON /PRTINT/ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,  FAN 0090
     -IRUD,IPROP,IAEROD,IRHS                                            FAN 0100
      COMMON/SOFTBS/XBF,PBS,SINBS,COSBS,XBS,ZBS,DELYBS,DPBS,ELMAXB,YAVG FAN 0110
     1B(10)                                                             FAN 0120
      COMMON /SOFTSS/ XLF,PSS,SINTH,COSTH,XSS,ZSS,DELYSS,DPSS           FAN 0130
     1 ,LLMAXS,YAVGS(10)                                                FAN 0140
      COMMON /VARBLE/ VAL(40)                                           FAN 0150
      DIMENSION QB(1),QM(1),QS(1),PBOW(1),PM(1),PS(1)                   FAN 0160
      EQUIVALENCE (VAL(1),TIME),(VAL(2),U),(VAL(3),V),(VAL(4),W),       FAN 0170
     1(VAL(5),P),(VAL(6),Q),(VAL(7),R),(VAL(8),PHI),(VAL(9),THETA),     FAN 0180
     2(VAL(10),Z),(VAL(11),BMASS),(VAL(21),X),(VAL(22),Y),(VAL(23),PSI),FAN 0190
     3(VAL(24),PB)                                                      FAN 0200
      EQUIVALENCE (VAL(18),FANPWR)                                      FAN 0210
```

```
      EQUIVALENCE (QBFAN(1),QB(1)),(QMFAN(1),QM(1)),(QSFAN(1),QS(1)),     FAN 0220
     1 (PBFAN(1),PBOW(1)),(PMFAN(1),PM(1)),(PSFAN(1),PS(1))               FAN 0230
                                                                          FAN 0240
      BRAT=3000/BRPM                                                      FAN 0250
      EMRAT=3000/EMRPM                                                    FAN 0260
      SRAT=3000/SRPM                                                      FAN 0270
      TL=VAL(1)                                                           FAN 0280
      IF(NB.EQ.0.0) GO TO 5                                               FAN 0290
      DPBS=FGI(TL,NB,TMEB,DELB,ILB)                                       FAN 0300
      PBS=PB+DPBS                                                         FAN 0310
    5 IF(NS.EQ.0.0) GO TO 6                                               FAN 0320
      DPSS=FGI(TL,NS,TMES,DELS,ILS)                                       FAN 0330
      PSS=PB+DPSS                                                         FAN 0340
    6 CONTINUE                                                            FAN 0350
      PB1=PBS-PINF                                                        FAN 0360
      PB2=PB-PINF                                                         FAN 0370
      PB3=PSS-PINF                                                        FAN 0380
      PBARB=PB1*BRAT**2                                                   FAN 0390
      PBARM=PB2*EMRAT**2                                                  FAN 0400
      PBARS=PB3*SRAT**2                                                   FAN 0410
                                                                          FAN 0420
      QBOW=ENBFAN*FG1(PBARB,NPTSB,PBOW,QB,IB)/BRAT                        FAN 0430
      QMAIN=ENMFAN*FG1(PBARM,NPTSM,PM,QM,IM)/EMRAT                        FAN 0440
      QSTN=ENSFAN*FG1(PBARS,NPTSS,PS,QS,IS)/SRAT                          FAN 0450
      QIN=QBOW+QMAIN+QSTN                                                 FAN 0460
      FANPWR=(QBOW*PB1+QMAIN*PB2+QSTN*PB3)/550.                           FAN 0470
      IF (IRHS.NE.ON) RETURN                                              FAN 0480
      WRITE(6,100) QBOW,QMAIN,QSTN,PBARB,PBARM,PBARS                      FAN 0490
  100 FORMAT(//4H FAN /32H Q - BOW,MAIN,STERN (CU FT /SEC) 3F12.1         FAN 0500
     1 /28H DELP - BOW,MAIN,STERN (PSF) 3F11.2)                           FAN 0510
      RETURN                                                              FAN 0520
      END                                                                 FAN 0530

C                                                                         FG1 0010
      FUNCTION FG1(X,N,XT,YT,IX)                                          FG1 0020
      DIMENSION XT(1),YT(1)                                               FG1 0030
      IF (IX.LT.1) IX=1                                                   FG1 0040
      IF (IX.GT.N-1) IX=N-1                                               FG1 0050
      I=SIGN(1.0,X-XT(IX))                                                FG1 0060
 10   IF (IX.LT.1 .OR. IX.GE.N) GO TO 30                                  FG1 0070
      IF (XT(IX).GT.X .OR. X.GT.XT(IX+1)) GO TO 20                        FG1 0080
      C=(X-XT(IX))/(XT(IX+1)-XT(IX))                                      FG1 0090
      GO TO 100                                                           FG1 0100
 20   IX=IX+I                                                             FG1 0110
      GO TO 10                                                            FG1 0120
 30   C=IX/N                                                              FG1 0130
      IX=IX-I                                                             FG1 0140
```

```
100   FG1=YT(IX)+C*(YT(IX+1)-YT(IX))
      RETURN
      END


C
      SUBROUTINE FORIT(FNT,N,M,A,B,IER)

      DIMENSION A(1),B(1),FNT(1)

C     CHECK FOR PARAMETER ERRORS
      IER=0
20    IF(M) 30,40,40
30    IER=2
      RETURN
40    IF(M-N) 60,60,50
50    IER=1
      RETURN

C     COMPUTE AND PRESET CONSTANTS
 60   CONTINUE
      AN=N
      COEF=2.0/(2.0*AN+1.0)
      PI=4.*ATAN(1.)
      CONST=PI*COEF
      S1=SIN(CONST)
      C1=COS(CONST)
      C=1.0
      S=0.0
      J=1
      FNTZ=FNT(1)
70    U2=0.0
      U1=0.0
      I=2*N+1

C     FORM FOURIER COEFFICIENTS RECURSIVELY
75    U0=FNT(I)+2.0*C*U1-U2
      U2=U1
      U1=U0
      I=I-1
      IF(I-1) 80,80,75
80    A(J)=COEF *(FNTZ+ C*U1-U2)
      B(J)=COEF*S*U1
      IF(J-(M+1)) 90,100,100
90    Q=C1*C-S1*S
      S=C1*S+S1*C
      C=C
      J=J+1
```

```
      GO TO 70
  100 A(1)=A(1)*0.5
      RETURN
      END

      SUBROUTINE INCON (TIME)
      REAL*8 TICRD
      INTEGER ON
      COMMON /AIR/ PINF,RHOINF,GAM
      COMMON/AXIS/NXYS(26)
      COMMON /BMCO / IMM,IMNX,IMNY,IBMFIL,BTIME,IMT,XMI(10),YMI(7),IX,IY
      COMMON /COLUMN/ IVERT,ILATRL
      COMMON /CONST/ PI,RAD,U0
      COMMON /CNTRL/CONTW,CONTQ,CONTTH,QMULT,LOUVER,ACONTZ,ACONTW,ZEQUIL
     1 ,THEQL,ACBASE
      COMMON /CURVE/NCURV(10)
      COMMON/ENGINE/NPS,NPP,THSTS(25),THSTP(25),XP,YP,ZP,STHS,STHP,
     ATIP(25),TIS(25)
      COMMON /EQNCO/ NEQS,TOL(20),JQQ
      COMMON /FAIR/ RHOA,XLAERO
      COMMON /FANMAP/QIN,QBFAN(25),QMFAN(25),QSFAN(25),ENBFAN,ENMFAN,
     1 ENSFAN,BRPM,EMRPM,SKPM,NPTSB,NPTSM,NPTSS
     2 ,PBFAN(25),PMFAN(25),PSFAN(25),TMEB(25),DELB(25),NB,TMES(25),
     3DETS(25),NS
      COMMON / FROUDE / FN,FNCRIT
      COMMON/GBOW/ XBOW
      COMMON /GEOM/    WIDTH,XL,XX(4,11),YY(4,11),NSTA(4),AB,VOLNOM
     1,DELS(4,10),XCP,ZCP
      COMMON /GEOMSW/ XAVG(10),DS
      COMMON/GRAF/NGRAF,NDRW
      COMMON /HEADG/TICRD(6)
      COMMON /PWAVE/ FNCON,PWVCON
      COMMON/LEAKER/ALEAK,BLEAK,CFSS,CFBS
      COMMON /MASSES/ AM,AIXX,AIYY,AIZZ,AIXZ,AIMAX,G,WEIGHT,RHO,NMASS,
     -               AMI(201),XI(201),YI(201),ZI(201),XS,ZS,HRHO
      COMMON /MATRIX/ A(6,6)
      COMMON /OPTION/ I3DOF,ISRGE,ITRIM,NVALID
      COMMON /PLENUM/XLBW,XBBW,ABW,BUBHGT
      COMMON/PLVCQQ/NVI,NVO,NLI,NLD
      COMMON /PRIME/ STIME,FTIME,DELT,DELPNT,TPRINT
      COMMON /PRTINT/ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,
     -IRUD,IPROP,IAEROD,IRHS
      COMMON /ROLL/ PHIMAX,TROLL
      COMMON/RUDDR/ NPR,DELRUD(25),XR,YR,ZR,IRDS,TL,RSPAN,RAREA,RASPR,
     ARCLB,RTC,RUDANG,TIR(25)
      COMMON /RISER/ AMPTC
```

```
      COMMON /SLOPE/WATSLP,XPWV,XLXPWV,XPWVXS                           INCN0420
      COMMON/SOFTBS/XBF,PBS,SINBS,COSBS,XBS,ZBS,DELYBS,DPBS,ELMAXB,YAVG INCN0430
     1B(10)                                                             INCN0440
      COMMON /SOFTSS/ XLF,PSS,SINTH,COSTH,XSS,ZSS,DELYSS,DPSS           INCN0450
     1 ,ELMAXS,YAVGS(10)                                                INCN0460
      COMMON /SIDE/FXSW,FYSW,FZSW,FKSW,FMSW,FNSW,ALSW,YSW,XLSW,CFSW,CDSWINCN0470
     1,VAREA  ,VCHORD,VSPAN,VANGLE,VCOS,VX,VY,VZ  ,AVBMSW,DELX,VTC      INCN0480
      COMMON /STABLE/ S(4),ISTAB                                        INCN0490
      COMMON /STSLR/ CPHI,CPHID                                         INCN0500
      COMMON /SUM/ ISUM1(3),ISUM2(8)                                    INCN0510
      COMMON / VALOLD / YOLD(20)                                        INCN0520
      COMMON /VARBLE/ VAL(40)                                           INCN0530
      COMMON /WAVE/ ETA(4,11),AW(10),OMEGA(10),DVOLW,NWAVE,BETA,        INCN0540
     1              FXWAV,FYWAV,FZWAV,FKWAV,FMWAV,FNWAV                 INCN0550
     2              ,ZBAR,PHIBAR,THEBAR,TC,COSBET,SINBET,PBBAR          INCN0560
      COMMON /WAVEF/WAVLEN(10),OMEGAE(10),WAVSLP(10),ENCPER(10)         INCN0570
      COMMON /WAVTAB/ NAL,DAL,SAL,NDS,DDS,SDS,NTH,DTH,STH,NBB,DBB,SBB,  INCN0580
     1          AC1(20,5,7),AC2(20,5,7),AC3(20,5,7),AC4(20,5,7),        INCN0590
     2          AC5(20,5,7),AC6(20,5,7),AC7(20,5,7)                     INCN0600
     3          ,AC0(20,5,7),AC00(20,5,7),AC8(20,5,7),                  INCN0610
     4          AS1(20,5,7),AS2(20,5,7),AS3(20,5,7),AS4(20,5,7),        INCN0620
     5          AS5(20,5,7),AS6(20,5,7),AS7(20,5,7)                     INCN0630
     6          ,AS0(20,5,7),AS00(20,5,7),AS8(20,5,7)                   INCN0640
     7          ,BB(36),XREF,RX                                         INCN0650
                                                                        INCN0660
      DIMENSION ZZZ(14050)                                              INCN0670
      EQUIVALENCE (ZZZ,NAL)                                             INCN0680
      EQUIVALENCE                (VAL(2),U),(VAL(3),V),(VAL(4),W),      INCN0690
     1(VAL(5),P),(VAL(6),Q),(VAL(7),R),(VAL(8),PHI),(VAL(9),THETA),     INCN0700
     2(VAL(10),Z),(VAL(11),BMASS),(VAL(21),X),(VAL(22),Y),(VAL(23),PSI),INCN0710
     3(VAL(24),PB)                                                      INCN0720
      DIMENSION TEMP(7),XMO(10)                                         INCN0730
      DIMENSION TITLC(20)                                               INCN0740
      DATA BEAM,BETAD,DELO,DELPI,DLRDO,DSO,ISYS,RMAXO,RONO,RRATO,RREVO, INCN0750
     1 THETO,THSSI,TPRINO,UO,VXO,VZO,XBSI,XCPO,XLTOT,XPO,XRO,XSSI,YPO,  INCN0760
     2 ZPO,ZRO,ZSSI/ 6*0.0,0,20*0.0/                                    INCN0770
      DO 9 I=1,8                                                        INCN0780
      ISUM1(I)=0                                                        INCN0790
    9 ISUM2(I)=0                                                        INCN0800
      PINF=2116.                                                        INCN0810
      RHOINF=.002378                                                    INCN0820
      GAM=1.4                                                           INCN0830
      GO TO 10                                                          INCN0840
 2200 READ(5,3000) NGRAF,NDRW                                           INCN0850
 3000 FORMAT(2I2)                                                       INCN0860
      READ(5,3001) NXYS                                                 INCN0870
 3001 FORMAT(26I2)                                                      INCN0880
      READ(5,3002) TICRD                                                INCN0890
```

```
 3002 FORMAT(6A8)
10    READ(5,99) ISYSL,IOPT,(TEMP(I),I=1,7)
      IF( ISYSL .EQ. ISYS .AND. ISYSL.EQ. 13) GOTO 70
      ISYS=ISYSL
      IF((ISYS.LE.0).OR.(ISYS.GT.22)) GO TO 70
      GOTO(100,200,300,400,500,600,700,800,900,1000,1100,1200,1300,
     11400,1500,1600,1700,1800,1900,2000,2100,2200),ISYS

C     PROGRAM CONTROL PARAMETERS
100   CONTINUE
      GOTO (101,102,103,104,105),IOPT
 101  CONTINUE
      STIME=TEMP(1)
      FTIME=TEMP(2)
      DELO=TEMP(3)
      DELPNT=TEMP(4)
      TPRINO=TEMP(5)
      IF (TPRINO.LT.STIME+DELPNT)  TPRINO = STIME+DELPNT
      IF (DELO.GT.DELPNT) DELO=DELPNT
      IF (DELO.EQ.0.0) GO TO 140
      GOTO 10
 2000 READ(5,3003) NCURV
 3003 FORMAT(10I1)
      GO TO 10
 2100 READ(5,2210) ISUM1
      READ(5,2210)ISUM2
 2210 FORMAT(8I2)
      GO TO 10
102   READ (5,191) IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,IRUD,
     1 IPROP,IAEROD,IRHS
      GOTO 10
103   READ (5,175) NEQS,JQQ,(TOL(J),J=1,NEQS)
      GOTO 10
 104  READ(5,191) IVERT,ILATRL,NVD,NVI,NLD,NLI
      GO TO 10
 105  CONTINUE
      I3DOF=TEMP(1)
      ISRGE=TEMP(2)
      ITRIM=TEMP(3)
      GO TO 10
  140 WRITE(6,195)
      STOP

C     MASS DISTRIBUTION
  200 G=32.17
      RHO=1.99
      HRHO=RHO/2.
      GO TO (210,220,230), IOPT
```

```
  210 IMM = 0                                                             INCN1380
      WEIGHT = TEMP(1)                                                    INCN1390
      AM = WEIGHT/G                                                       INCN1400
      XS = TEMP(2)                                                        INCN1410
      ZS = TEMP(3)                                                        INCN1420
      AIXX = TEMP(4)                                                      INCN1430
      AIYY = TEMP(5)                                                      INCN1440
      AIZZ = TEMP(6)                                                      INCN1450
      AIXZ = TEMP(7)                                                      INCN1460
C     INERTIA MATRIX OPERATIONS                                           INCN1470
  212 DO 211 I=1,6                                                        INCN1480
      DO 211 N=1,6                                                        INCN1490
  211 A(I,N) = 0.0                                                        INCN1500
      DO 213 N=1,3                                                        INCN1510
  213 A(N,N)=AM                                                           INCN1520
      A(4,4)=AIXX                                                         INCN1530
      A(5,5)=AIYY                                                         INCN1540
      A(6,6)=AIZZ                                                         INCN1550
      A(4,6)=-AIXZ                                                        INCN1560
      A(6,4)=-AIXZ                                                        INCN1570
      AIMAX=AMAX1(AM,AIXX,AIYY,AIZZ,ABS(AIXZ))                            INCN1580
      DO 214 I=1,6                                                        INCN1590
      DO 214 J=1,6                                                        INCN1600
  214 A(I,J)=A(I,J)/AIMAX                                                 INCN1610
      CALL DMINV (A,6,D)                                                  INCN1620
      DO 215 I=1,6                                                        INCN1630
      DO 215 J=1,6                                                        INCN1640
  215 A(I,J)=A(I,J)/AIMAX                                                 INCN1650
      IF (D.NE.0.0) GO TO 10                                              INCN1660
      WRITE (6,216)                                                       INCN1670
      STOP                                                                INCN1680
C     READ WEIGHT DISTRIBUTION - ASSUME TRANSVERSE (PORT/STBD) SYMMETRY   INCN1690
C        X INPUT DIST. FWD. OF (SIDEWALL) TRANSOM                         INCN1700
C        Y INPUT DIST. TO STARBOARD                                       INCN1710
  220 I = 1                                                               INCN1720
C        Z INPUT DIST. UP FROM KEEL-LINE                                  INCN1730
  222 READ (5,192) AMI(I),XI(I),YI(I),ZI(I)                               INCN1740
      IF (AMI(I).LT.0.0)  GO TO 224                                       INCN1750
      I = I+1                                                             INCN1760
      IF ( I.GT.201 )  GO TO 70                                           INCN1770
      GO TO 222                                                           INCN1780
  224 NMASS = I-1                                                         INCN1790
      SUM = 0.0                                                           INCN1800
      SUX = 0.0                                                           INCN1810
      SUZ = 0.0                                                           INCN1820
      DO 225 I=1,NMASS                                                    INCN1830
      AMI(I) = AMI(I)/G                                                   INCN1840
      SUM = SUM+AMI(I)                                                    INCN1850
```

```
      SUX =  SUX+AMI(I)*XI(I)
  225 SUZ =  SUZ+AMI(I)*ZI(I)
      AM = SUM*2.0
      WEIGHT  = AM*G
      ZS   =  SUZ/SUM
      XS   =  SUX/SUM
      SUM =  0.0
      SUX =  0.0
      SUY =  0.0
      SUZ =  0.0
      DO 226 I=1,NMASS
      XI(I)  =  XI(I)-XS
      ZI(I)  =  -ZI(I)+ZS
      AMK =  AMI(I)*2.0
      SUX =  SUX+AMK*XI(I)*XI(I)
      SUY =  SUY+AMK*YI(I)*YI(I)
      SUZ =  SUZ+AMK*ZI(I)*ZI(I)
  226 SUM =  SUM+AMK*XI(I)*ZI(I)
      AIXX =  SUY+SUZ
      AIYY =  SUX+SUZ
      AIZZ =  SUX+SUY
      AIXZ =  SUM
      GO TO 212
  230 GO TO 10

C     XX AND YY TABLES
300   CONTINUE
      NSTA(1)  =  TEMP(1)
      NSTA(2)  =  TEMP(2)
      NSTA(3)  =  TEMP(3)
      NSTA(4)  =  TEMP(4)
      XLTOT=TEMP(5)
      GOTO 10

C     SIDEWALL (INCLUDING APPENDAGES)
400   CONTINUE
      GOTO (401,402),IOPT
401   YSW=TEMP(1)
      XLSW=TEMP(2)
      CFSW=TEMP(3)
      CDSW=TEMP(4)
      AVBMSW=TEMP(5)
      READ (10) ZZZ
      REWIND 10
      GOTO 10
C    BLOCK 4 OPTION 2 REMOVED.       NO APENDAGES
  402 CONTINUE
      GOTO 10
```

```
C       STERNSEAL
500     CONTINUE
        XSSI=TEMP(1)
        ZSSI=TEMP(2)
        ALEAK=TEMP(3)
        CFSS=TEMP(4)
        THSSI=TEMP(5)
        DPSS=TEMP(6)
        XLF=TEMP(7)
        SINTH=SIN(THSSI/RAD)
        COSTH=COS(THSSI/RAD)
        ELMAXS=XLF*COSTH
        GO TO 10

C       BOWSEAL
600     CONTINUE
        XBSI=TEMP(1)
        CFBS=TEMP(2)
        DPBS=TEMP(3)
        ZBSI=TEMP(4)
        THBSI=TEMP(5)
        XBF=TEMP(6)
        BLEAK=TEMP(7)
        SINBS=SIN(THBSI/RAD)
        COSBS=COS(THBSI/RAD)
        ELMAXB=XBF*COSBS
        GOTO 10

C       PLENUM
700     CONTINUE
        GO TO (705,710),IOPT
 705    CONTINUE
        XLBW=TEMP(1)
        XBBW=TEMP(2)
        XPWV=TEMP(3)
        WIDTH=TEMP(4)
        XL=TEMP(5)
        XCPU=TEMP(6)
        BUBHGT=TEMP(7)
        XLXPWV=XLBW-XPWV
        XPWVXS=XPWV-XS
        ABW=XBBW*XLBW
        AB=WIDTH*XL
        VCLNUM=(ABW+AB)*BUBHGT/2.
        GOTO 10
 710    CONTINUE
        FNCRIT=TEMP(1)
```

```
      GO TO 10

C     PROPULSION
800   CONTINUE
      GO TO (805,810),IOPT
 805  CONTINUE
      XPO=TEMP(1)
      YPO=TEMP(2)
      ZPO=TEMP(3)
      GO TO 10
C BLOCK 8 OPTION 2 REMOVED. ENGINE OUT INPUT IN BLOCK 16
  810 CONTINUE
      GOTO 10

C     RUDDER
900   CONTINUE
      GO TO (905,910,915),IOPT
  905 XRO = TEMP(1)
      YR=TEMP(2)
      ZRO = TEMP(3)
      RSPAN=TEMP(4)
      RASPR=TEMP(5)
      RAREA=TEMP(6)
      RCLB=2.*PI*RASPR/(RASPR+3.)
      RTC=TEMP(7)
      GO TO 10
C     910 NOT USED
  910 CONTINUE
      GO TO 10
 915  CONTINUE
      GOTO 10

C     AERODYNAMICS
1000  CONTINUE
      XLAERO=TEMP(1)
      BEAM=TEMP(2)
      RHOA=.5*RHOINF*XLAERO*BEAM
      GOTO 10

C     WAVES
1100  CONTINUE
      IWAVSW=IOPT
      NWAVE=TEMP(1)
      IF(NWAVE .EQ. 0) GOTO 10
      IF(NWAVE .GT. 10) GOTO 70
      BETAD=TEMP(2)
      BETA=BETAD/RAD
      COSBET=COS(BETA)
```

```
      SINBET=SIN(BETA)
      TC = 1.0
      GOTO ( 1104,1106) ,IWAVSW
1104  DO 1105 I=1,NWAVE
1105  READ(5,1190) OMEGA(I),AW(I)
      GOTO 10
1106  DO 1107 I=1,NWAVE
1107  READ (5,1190) WAVLEN(I) ,AW(I)
      GOTO 10

C     INITIAL CONDITIONS
1200  CONTINUE
      UO = TEMP(1)
      THETO = TEMP(2)
      DSO = TEMP(3)
      DELPI=TEMP(4)
      GO TO 10

1300  CONTINUE
C     INPUT COMPLETED. 1) PRINT ALL INPUT
      WRITE(6,2004) TITLC
      WRITE (6,2001) STIME,FTIME,DELO,TPRINO,DELPNT
      WRITE (6,2002) IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,IRUD,
     1 IPROP,IAEROD,IRHS
      WRITE(6,2021) I3DOF,ISRGE,ITRIM,NVALID
      WRITE (6,2003) NEQS, (TOL(J),J=1,NEQS)
      WRITE (6,219) WEIGHT,XS,ZS,AIXX,AIYY,AIZZ,AIXZ
      WRITE(6,217)A,AIMAX
      WRITE (6,2018) NSTA
      WRITE (6,490) YSW,XLSW,CFSW,CDSW,VANGLE,VSPAN,VCHORD,VXO,VY,VZO,
     1 AVBMSW,VTC
      WRITE (6,491) NAL,DAL,SAL,NDS,DDS,SDS,NTH,DTH,STH,NBB,DBB,SBB
      IF (IMM.GT.0)  WRITE(6,1549)   (XMD(J),J=1,IMNX)
      WRITE (6,1519) IMM,IMNX,IMNY,IBMFIL,BTIME,IMT
      IF (IMM.GT.0)  WRITE(6,1559) (YMI(J),J=1,IMNY)
      WRITE (6,2010) XLBW,XBBW
      WRITE (6,2011) XL,WIDTH,XCPO,VOLNOM,BUBHGT
      WRITE (6,2020) DELPI
      WRITE(6,2009) FNCRIT,XLTOT
      WRITE (6,2028) ENBFAN,BRPM,ENMFAN,EMRPM,ENSFAN,SRPM
      WRITE (6,2013) XRO,YR,ZRO,RONO,RMAXO,RRATO,RREVO,DLRDO
     1   ,RSPAN,RASPR,RAREA,RCLB,RTC
      WRITE (6,2012)       XPO,YPO,ZPO
      WRITE (6,2027) XLAERO,BEAM
      WRITE (6,2026) XBSI,CFBS,DPBS,ELMAXB
      WRITE (6,2025) XSSI,ZSSI,ALEAK,CFSS,THSSI,DPSS,XLF
      WRITE (6,2017) UO,THETO,DSO
```

```
C                     AND 2) INITIALIZE VARIABLES FOR CALCS.
      DO 1302 I=1,40
 1302 VAL(I) = 0.0
      U = U0*1.6878
      XSS=-(XS-XSSI)
      ZSS=ZS-ZSSI
      THETA = THETO/RAD
      THEQL=THETA
      DS = DS0/12.
      Z=-ZS+DS
      ZEQUIL=Z
      PHIMAX=0.
      TROLL=0.
      IRDS=0
      TL=0.0
C     WAVE PARAMETERS TABLE
      IF(NWAVE .EQ.0) GOTO 1321
      AMPTC=1.30287
      GOTO(1310,1315),IWAVSW
1310  DO 1311 I=1,NWAVE
1311  WAVLEN(I)=2.*PI*G/(OMEGA(I)*OMEGA(I))
      GOTO 1317
1315  DO 1316 I=1,NWAVE
1316  OMEGA(I)=SQRT(2.*PI*G/WAVLEN(I))
1317  CONTINUE

C     CALCULATE INITIAL FREQUENCIES OF ENCOUNTER
      DO 1318 I=1,NWAVE
      WAVSLP(I) = 360.0*AW(I)/WAVLEN(I)
      OMEGAE(I) =2.*PI*(SQRT(G*WAVLEN(I)/(2.*PI))-U*COSBET)/WAVLEN(I)
      ENCPER(I) = 2.0*PI/OMEGAE(I)
1318  CONTINUE
      WRITE (6,1191) NWAVE,BETAD,(OMEGA(I),OMEGAE(I),WAVLEN(I),AW(I),
     1                WAVSLP(I),ENCPER(I),I=1,NWAVE)
      GOTO 1322
 1321 WRITE (6,1192)
1322  CONTINUE
      DO 1303 I=1,4
      DO 1303 N=1,11
 1303 ETA(I,N) = 0.0
      DVOLW = 0.0
      FXWAV = 0.0
      FYWAV = 0.0
      FZWAV = 0.0
      FKWAV = 0.0
      FMWAV = 0.0
      FNWAV = 0.0
      ZBAR=Z
```

```
      PHIBAR=PHI                                                INCN4260
      THEBAR=THETA                                              INCN4270
      TIME=STIME                                                INCN4280
      DELT = DELO                                               INCN4290
      TPRINT=TPRINO-DELPNT                                      INCN4300
      PWVCON=4.*WEIGHT/(RHO*G*XLBW)                             INCN4310
      FNCON=SQRT(XLBW*G)                                        INCN4320
      VX=VXO-XS                                                 INCN4330
      VZ = ZS-VZO                                               INCN4340
      XP=XPO-XS                                                 INCN4350
      XR = XRO-XS                                               INCN4360
      YP=YPO                                                    INCN4370
      ZP=ZS-ZPO                                                 INCN4380
      ZR = ZS-ZRO                                               INCN4390
      IF (IMM.EQ. 0) GO TO 1305                                 INCN4400
      DO 1304 J=1,IMNX                                          INCN4410
 1304 XMI(J) = XMO(J) - XS                                      INCN4420
 1305 CONTINUE                                                  INCN4430
      XCP = XCPO-XS                                             INCN4440
      ZCP = ZS-BUBHGT                                           INCN4450
      XBS=XBSI-XS                                               INCN4460
      N=NSTA(3)                                                 INCN4470
      ZBS=ZS-ZBSI                                               INCN4480
      DO 1364 J=1,N                                             INCN4490
      DELYBS=XBBW/(N-1)                                         INCN4500
       XX(3,J)=XBS-XSSI                                         INCN4510
      YY(3,J)=-0.5*XBBW+(J-1)*DELYBS                            INCN4520
 1364 CONTINUE                                                  INCN4530
      N=N-1                                                     INCN4540
      DO 1367 J=1,N                                             INCN4550
      YAVGB(J)=(YY(3,J+1)+YY(3,J))/2.                           INCN4560
1367  CONTINUE                                                  INCN4570
      N=NSTA(4)                                                 INCN4580
      DELYSS=XBBW/(N-1)                                         INCN4590
      DO 1365 J=1,N                                             INCN4600
      XX(4,J)=-XS                                               INCN4610
      YY(4,J)=-.5*XBBW+(J-1)*DELYSS                             INCN4620
 1365 CONTINUE                                                  INCN4630
      N=N-1                                                     INCN4640
      DO 1368 J=1,N                                             INCN4650
      YAVGS(J)=(YY(4,J+1)+YY(4,J))/2.                           INCN4660
 1368 CONTINUE                                                  INCN4670
      XBOW=XLTOT-XS                                             INCN4680
      N=NSTA(1)                                                 INCN4690
      DELX=XBSI/(N-1)                                           INCN4700
      DO 1309 J=1,2                                             INCN4710
      DO 1309 I=1,N                                             INCN4720
      XX(J,I)=(I-1)*DELX-XS                                     INCN4730
```

```
1309  YY(J,I)= YSW*(2*J-3)                                                 INCN4740
      WRITE(6,1366) ((XX(J,N),N=1,11),(YY(J,N),N=1,11),J=1,4)              INCN4750
 1366 FORMAT (//17H XX AND YY ARRAYS /14H PORT SIDEWALL /2(11F10.2/),      INCN4760
     1 15H STBD. SIDEWALL /2(11F10.2/),9H BOW SEAL /2(11F10.2/),           INCN4770
     2 11H STERN SEAL /2(11F10.2/))                                        INCN4780
      N=NSTA(1)-1                                                          INCN4790
      DO 1308 I=1,N                                                        INCN4800
1308  XAVG(I)=DELX*(2*I-1)/2.-XS                                           INCN4810
      CALL WAVES(TIME)                                                     INCN4820
                                                                           INCN4830
C     INITIALIZE BUBBLE PRESSURE, ABSOLUTE (PSF)                           INCN4840
      PB=PINF+DELPI                                                        INCN4850
      PBBAR=DELPI                                                          INCN4860
      PBAR=DELPI                                                           INCN4870
      PSS=PB+DPSS                                                          INCN4880
      PBS=PB+DPBS                                                          INCN4890
      AB=ABW-(ABW-AB)*(ZS+Z/BUBHGT)                                        INCN4900
      CF=.37/((U/FNCON)**1.5655981)                                        INCN4910
      WATSLP=PBBAR*CF*PWVCON/WEIGHT                                        INCN4920
      VOL=VOLNOM-.5*(AB+ABW)*(Z+ZS)-DVOLW                                  INCN4930
     1+.5*WATSLP*XL*AB                                                     INCN4940
      BMASS=(PB/PINF)**(1./GAM)*VOL*RHOINF                                 INCN4950
      WRITE (6,2023)                                                       INCN4960
      RETURN                                                               INCN4970
                                                                           INCN4980
C     RUN TERMINATOR                                                       INCN4990
 1400 WRITE(6,98)                                                          INCN5000
      STOP                                                                 INCN5010
                                                                           INCN5020
C     BENDING MOMENT                                                       INCN5030
 1500 GO TO (1510,1520,1530,1540), IOPT                                    INCN5040
 1510 IMM = TEMP(1)                                                        INCN5050
      IF (IMM.GT.3)  GO TO 70                                              INCN5060
      IMNX = TEMP(2)                                                       INCN5070
      IF (IMNX.GT.10)  GO TO 70                                            INCN5080
      IMNY = TEMP(3)                                                       INCN5090
      IF (IMNY.GT.7)  GO TO 70                                             INCN5100
      IBMFIL = TEMP(4)                                                     INCN5110
      BTIME = TEMP(5)                                                      INCN5120
      IF (IMM.EQ.3)  IMT = TEMP(6)                                         INCN5130
      GO TO 10                                                             INCN5140
 1520 DO 1521 J=1,7                                                        INCN5150
 1521 XMO(J) = TEMP(J)                                                     INCN5160
      IF (IMNX.LE.7)  GO TO 10                                             INCN5170
      READ 1522, (XMO(J),J=8,IMNX)                                         INCN5180
      GO TO 10                                                             INCN5190
 1530 DO 1531 J=1,IMNY                                                     INCN5200
 1531 YMI(J) = TEMP(J)                                                     INCN5210
```

```
      GO TO 10                                                          INCN5220
 1540 CONTINUE                                                          INCN5230
      GO TO 10                                                          INCN5240
                                                                        INCN5250
1600  CONTINUE                                                          INCN5260
      GO TO(1605,1610,1615),IOPT                                        INCN5270
 1605 CONTINUE                                                          INCN5280
C   VALUES INPUT FOR STBD SCREW                                         INCN5290
      THST1=TEMP(1)                                                     INCN5300
      NPS=TEMP(2)                                                       INCN5310
      STHS=TEMP(3)                                                      INCN5320
      IF(NPS.EQ.0.0) GO TO 1609                                         INCN5330
      READ(5,1950)(TIS(J),J=1,NPS)                                      INCN5340
      READ(5,1950)(THSTS(J),J=1,NPS)                                    INCN5350
      GO TO 10                                                          INCN5360
 1609 THSTS(1)=THST1                                                    INCN5370
 1610 CONTINUE                                                          INCN5380
C   VALUES INPUT FOR PORT SCREW                                         INCN5390
      THST2=TEMP(1)                                                     INCN5400
      NPP=TEMP(2)                                                       INCN5410
      STHP=TEMP(3)                                                      INCN5420
      IF(NPP.EQ.0.0) GO TO 1614                                         INCN5430
      READ(5,1950)(TIP(J),J=1,NPP)                                      INCN5440
      READ(5,1950)(THSTP(J),J=1,NPP)                                    INCN5450
      GO TO 10                                                          INCN5460
 1614 THSTP(1)=THST2                                                    INCN5470
 1615 CONTINUE                                                          INCN5480
C   VALUES INPUT FOR RUDDER                                             INCN5490
      DELR=TEMP(1)                                                      INCN5500
      NPR=TEMP(2)                                                       INCN5510
      IF(NPR.EQ.0.0) GO TO 1616                                         INCN5520
      READ(5,1950)(TIR(J),J=1,NPR)                                      INCN5530
      READ(5,1950)(DELRUD(J),J=1,NPR)                                   INCN5540
      GO TO 10                                                          INCN5550
 1616 DELRUD(1)=DELR                                                    INCN5560
      GO TO 10                                                          INCN5570
                                                                        INCN5580
 1700 GO TO (1705,1710),IOPT                                            INCN5590
 1705 NB=TEMP(1)                                                        INCN5600
      READ(5,1950)(TMEB(I),I=1,NB)                                      INCN5610
      READ(5,1950)(DELB(I),I=1,NB)                                      INCN5620
      GO TO 10                                                          INCN5630
 1710 NS=TEMP(1)                                                        INCN5640
      READ(5,1950)(TMES(I),I=1,NS)                                      INCN5650
      READ(5,1950)(DETS(I),I=1,NS)                                      INCN5660
      GCTO 10                                                           INCN5670
                                                                        INCN5680
C     TITLE CARD (ALL 80 COLUMNS )                                      INCN5690
```

```
 1800 READ  (5,2022)  TITLC                                          INCN5700
      GO TO 10                                                       INCN5710
                                                                     INCN5720
C     FAN MAPS                                                       INCN5730
1900  CONTINUE                                                       INCN5740
      GO TO (1905,1910,1915),IOPT                                    INCN5750
1905  CONTINUE                                                       INCN5760
      ENBFAN=TEMP(1)                                                 INCN5770
      BRPM=TEMP(2)                                                   INCN5780
      NPTSB=TEMP(3)                                                  INCN5790
      READIN=TEMP(4)                                                 INCN5800
      IF (READIN .EQ. 0.0) GO TO 10                                  INCN5810
      READ (5,1950) (PBFAN(J),J=1,NPTSB)                             INCN5820
      READ (5,1950) (QBFAN(J),J=1,NPTSB)                             INCN5830
      GO TO 10                                                       INCN5840
1910  CONTINUE                                                       INCN5850
      ENMFAN=TEMP(1)                                                 INCN5860
      EMRPM=TEMP(2)                                                  INCN5870
      NPTSM=TEMP(3)                                                  INCN5880
      READIN=TEMP(4)                                                 INCN5890
      IF (READIN .EQ. 0.0) GO TO 10                                  INCN5900
      READ (5,1950) (PMFAN(J),J=1,NPTSM)                             INCN5910
      READ (5,1950) (QMFAN(J),J=1,NPTSM)                             INCN5920
      GO TO 10                                                       INCN5930
1915  CONTINUE                                                       INCN5940
      ENSFAN=TEMP(1)                                                 INCN5950
      SRPM=TEMP(2)                                                   INCN5960
      NPTSS=TEMP(3)                                                  INCN5970
      READIN=TEMP(4)                                                 INCN5980
      IF (READIN .EQ. 0.0) GO TO 10                                  INCN5990
      READ (5,1950) (PSFAN(J),J=1,NPTSS)                             INCN6000
      READ (5,1950) (QSFAN(J),J=1,NPTSS)                             INCN6010
      GO TO 10                                                       INCN6020
1950  FORMAT(8F10.0)                                                 INCN6030
C     ERROR IN INPUT                                                 INCN6040
   70 CONTINUE                                                       INCN6050
      WRITE (6,79) ISYS                                              INCN6060
      STOP                                                           INCN6070
   79 FORMAT(34H INPUT ERROR - - - STOP - -  ISYS= ,I3)              INCN6080
   98 FORMAT(1H1,20(/),50X,19H COMPLETED ALL RUNS )                  INCN6090
99    FORMAT(I3,I2,7F10.0)                                           INCN6100
 191  FORMAT (16I5)                                                  INCN6110
 192  FORMAT (5F10.0)                                                INCN6120
  195 FORMAT(///10X,65HERROR IN INPUT --- DELT AND/OR DELPNT EQUALS ZEROINCN6130
     1 --- JOB ABORTED )                                             INCN6140
 175  FORMAT (2I2/(8F10.0))                                          INCN6150
  216 FORMAT(///10X,82HERROR IN INPUT --- INPUT INERTIA ELEMENTS LEAD TOINCN6160
     1 ZERO DETERMINANT --- JOB ABORTED )                            INCN6170
```

```
  217 FORMAT(22H INERTIA MATRIX, AIMAX  6E15.4/(22X,6E15.4))             INCN6180
  219 FORMAT(30H WEIGHT, C.G., INERTIA MOMENTS 7F12.3)                    INCN6190
  305 FORMAT ( 11F7.0 )                                                   INCN6200
  490 FORMAT(15H SIDEWALL INPUT 12(F8.3,1X))                              INCN6210
  491 FORMAT(26H SIDEWALL TABLE PARAMETERS 4(I4,F7.3,F7.3))               INCN6220
 1190  FORMAT(2F10.0)                                                     INCN6230
 1191 FORMAT(/12HONO OF WAVES I2,10H BETA(DEG)F5.0/15H OMEGA(RAD/SEC) ,   INCN6240
     15X,16H OMEGAE(RAD/SEC) 5X,16H WAVE LENGTH(FT) 5X,14H AMPLITUDE(FT)INCN6250
     - 5X,16HMAX  SLOPE (DEG)5X,13HPERIOD,E(SEC)/(F8.4,12X,F8.4,4F20.3))INCN6260
 1192 FORMAT(/11HOCALM WATER)                                             INCN6270
 1519 FORMAT(32HOMOMENT CALC. CONTROL PARAMETERS 4I5, F8.3, I5 )          INCN6280
 1522 FORMAT ( 5X,7F10.0)                                                 INCN6290
 1549 FORMAT(22H MOMENT CALCS. AT X OF 11F10.3)                           INCN6300
 1559 FORMAT(22H MOMENT CALCS. AT Y OF 11F10.3)                           INCN6310
 2004 FORMAT(33H1SES MOTIONS AND LOADS PROGRAM - 20A4,/)                  INCN6320
 2001 FORMAT(23H START AND FINISH TIMES 2F10.2/                           INCN6330
     -22H INITIAL TIME INTERVAL F12.4/                                    INCN6340
     -18H START PRINTING AT F8.2,17H IN INCREMENTS OF F8.2)               INCN6350
 2002 FORMAT(24H INTERMEDIATE PRINT TAGS 16I5)                            INCN6360
 2003 FORMAT(39H NO. OF STATE EQUATIONS, AND TOLERANCESI5/(10X,10E12.2))INCN6370
 2009 FORMAT(23HOCRITICAL FROUDE NUMBER F15.4,5X,19H TOTAL CRAFT LENGTH INCN6380
     1          F15.4)                                                    INCN6390
 2010 FORMAT(34HOPLENUM, LENGTH AND WIDTH AT WATER 2F12.4)                INCN6400
 2011 FORMAT(34H PLENUM, LENGTH AND WIDTH AT HULL  2F12.4/                INCN6410
     -          35H PLENUM, CENTER OF PRESSURE AT HULL F12.4/             INCN6420
     -23H PLENUM, NOMINAL VOLUME F12.1,10X, 6HHEIGHT F12.4)               INCN6430
 2012 FORMAT(/33H PROPULSION, X, Y, Z COORDINATES  3F12.4/)               INCN6440
 2013 FORMAT(/28HORUDDER, X, Y, Z COORDINATES  3F12.4/                    INCN6450
     -           41H RUDDER, ON, MAX, RATE, REVERSE, INITIAL  5F12.4/     INCN6460
     -           33H RUDDER,SPAN,ASPECT,AREA,CLB,T/C  5F12.4)             INCN6470
 2017 FORMAT(/39HOINITIAL CONDITIONS, VELOCITY (KNOTS) = F7.2, 5X,        INCN6480
     -           13HPITCH (DEG) = F8.3,5X,12HDRAFT (IN) = F8.2)           INCN6490
 2018 FORMAT( 49H NUMBER OF STATIONS, SIDEWALLS (P+S), SEALS (B+S) ,4I5)INCN6500
 2020 FORMAT( 38H PLENUM, INITIAL PRESSURE, GAGE (PSF)  F8.2)             INCN6510
 2021 FORMAT(79H PROGRAM OPTION SWITCH SETTINGS (LATERAL PLANE,CONSTANT INCN6520
     -SPEED,TRIM,VALIDATION)   4I5)                                       INCN6530
 2022 FORMAT (20A4 )                                                      INCN6540
 2023 FORMAT ( 1H1 )                                                      INCN6550
 2025 FORMAT( 16H STERNSEAL INPUT  7F12.4 )                               INCN6560
 2026 FORMAT( 16H BOWSEAL INPUT    7F12.4 )                               INCN6570
 2027 FORMAT( 19HOAERODYNAMICS INPUT  7F12.4)                             INCN6580
 2028 FORMAT( 33HOFANS, NO. + RPM, BOW,MAIN,STERN  3(F10.0,F10.1))        INCN6590
      END                                                                 INCN6600

      SUBROUTINE INTGRL (TIME)                                            INT 0020
      INTEGER ON                                                          INT 0030
      COMMON /6MCO / IMM,IMNX,IMNY,IBMFEL,BTIME,IMT,XMI(10),YMI(7),IX,IYINT 0040
```

```
      COMMON /EQNCO/ NEQS,TOL(20),JQQ                                   INT 0050
      COMMON /KSWTCH/ ITHRST                                            INT 0060
      COMMON /MASSES/ AM,AIXX,AIYY,AIZZ,AIXZ,AIMAX,G,WEIGHT,RHO,NMASS,  INT 0070
     -                AMI(201),XI(201),YI(201),ZI(201),XS,ZS,HRHO       INT 0080
      COMMON /PRIME/ STIME,FTIME,DELT,DELPNT,TPRINT                     INT 0090
      COMMON /PRTINT/ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,  INT 0100
     -IRUD,IPROP,IAEROD,IRHS                                            INT 0110
      COMMON /STABLE/ S(4),ISTAB                                        INT 0120
      COMMON/STEP/ STEP2                                                INT 0130
      COMMON / VALOLD / YOLD(20)                                        INT 0140
      COMMON /VARBLE/ VAL(40)                                           INT 0150
      EQUIVALENCE (VAL(1),X),(VAL(2),Y(1))                              INT 0170
      DIMENSION Y(20),ERROR(20)                                         INT 0180
      REAL K1(20),K2(20),K3(20),K4(20),K5(20)                           INT 0190
      DATA IPASS/0/                                                     INT 0200
                                                                        INT 0210
      STEP2=1.0                                                         INT 0220
      PB=VAL(24)
      BMASS=Y(10)
    1 IF((TIME+DELT).LE.TPRINT) GO TO 12                                INT 0230
      DEL=DELT                                                          INT 0240
      DELT=TPRINT-TIME                                                  INT 0250
      IPASS=1                                                           INT 0260
   12 X=TIME                                                            INT 0270
      DO 2 J=1,NEQS                                                     INT 0280
      Y(J)=YOLD(J)                                                      INT 0290
    2 CONTINUE                                                          INT 0300
      ITHRST=1                                                          INT 0310
      CALL RHS(K1)                                                      INT 0320
      ITHRST=2                                                          INT 0330
      IMT = 0                                                           INT 0340
      IF (IACCEL.NE. ON) GO TO 14                                       INT 0350
      ACCLAT = (K1(2)+Y(1)*Y(6))/G                                      INT 0360
      WRITE  (6,101)  ACCLAT , DELT                                     INT 0370
   14 ON=2                                                              INT 0380
   15 H=DELT/3.                                                         INT 0390
      X=TIME+H                                                          INT 0400
      DO 3 J=1,NEQS                                                     INT 0410
    3 Y(J)=YOLD(J)+H*K1(J)                                              INT 0420
      CALL RHS(K2)                                                      INT 0430
      DO 4 J=1,NEQS                                                     INT 0440
    4 Y(J)=YOLD(J)+.5*H*(K1(J)+K2(J))                                   INT 0450
      CALL RHS(K3)                                                      INT 0460
      X=TIME+.5*DELT                                                    INT 0470
      DO 5 J=1,NEQS                                                     INT 0480
    5 Y(J)=YOLD(J)+.375*H*(K1(J)+3.*K3(J))                              INT 0490
      CALL RHS(K4)                                                      INT 0500
      X=TIME+DELT                                                       INT 0510
```

```
      DO 6 J=1,NEQS
    6 Y(J)=YOLD(J)+.5*H*(3.*K1(J)-9.*K3(J)+12.*K4(J))
      CALL RHS(K5)
      IF (JQQ.EQ. 1 ) GO TO 7
      DO 7 J=1,NEQS
      ERROR(J)=(K1(J)-4.5*K3(J)+4.*K4(J)-.5*K5(J))*H/5.0
      IF(ABS(ERROR(J)).GT.TOL(J)) GO TO 11
    7 CONTINUE
      DO 105 J=1,NEQS
      Y(J)=YOLD(J)+.5*H*(K1(J)+4.*K4(J)+K5(J))
  105 YOLD(J)=Y(J)
      TIME=TIME+DELT
      IF (IPASS.EQ.1) GO TO 8
      IF (JQQ.EQ. 1 ) GO TO 10
      DO 75 J=1,NEQS
      IF(ABS(ERROR(J)).GT.TOL(J)/16.) GO TO 9
   75 CONTINUE
      DELT=2.*DELT
      IF (DELT.GT.DELPNT) DELT=DELPNT
   10 RETURN
    9 STEP2=DELT
      GO TO 10
    8 DELT=DEL
      IPASS=0
      GO TO 10
   11 DELT=DELT/2.
      IF (DELT.LT. 1.E-6 )  GO TO 25
      IF(JQQ.EQ.2) GO TO 26
                      WRITE (6,666) TIME,DELT,J,ERROR(J),TOL(J)
   27 IPASS=0
      GO TO 15
   26 STEP1=DELT*2.0
      IF(STEP1.LT.STEP2)STEP2=STEP1
      GO TO 27
   25 WRITE ( 6,150 )
      WRITE (6,100) TIME,DELT,(K1(J),J=1,NEQS),VAL
      STOP
  100 FORMAT(/10X,23HINTGRL TIME,DELT,K1,VAL /2E15.4/2(5E15.4/),5(8E15.4
     1/))
  101 FORMAT(1H0,9X,33HTOTAL LATERAL ACCELERATION (G) =    F12.4,
     112X,5HDT =   E15.4)
  150 FORMAT(1H1,10X,44HDELTA TIME LESS THAN 1.0E-6  -  -  JOB STOPS )
  666 FORMAT(/10X,5HINT-J 2E30.5,I5,2E20.5)
      END
```

```
      SUBROUTINE PROP
      INTEGER ON
      COMMON /CONST/ PI,RAD,U0
      COMMON /FPROP/ FX,FY,FZ,FK,FM,FN
      COMMON/ENGINE/NPS,NPP,THSTS(25),THSTP(25),XP,YP,ZP,STHS,STHP,
     ATIP(25),TIS(25)
      COMMON /PRTINT/ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,
     -IRUD,IPROP,IAERUD,IRHS
      COMMON/RUDDR/ NPR,DELRUD(25),XR,YR,ZR,IRDS,TL,RSPAN,RAREA,RASPR,
     ARCLB,RTC,RUDANG,TIR(25)
      COMMON /VARBLE/ VAL(40)
      EQUIVALENCE (VAL(1),TIME),(VAL(2),U),(VAL(3),V),(VAL(4),W),
     1(VAL(5),P),(VAL(6),Q),(VAL(7),R),(VAL(8),PHI),(VAL(9),THETA),
     2(VAL(10),Z),(VAL(11),BMASS),(VAL(21),X),(VAL(22),Y),(VAL(23),PSI),
     3(VAL(24),PB)
      DIMENSION THS(1),THP(1),TS(1),TP(1),RUD(1),TR(1)
      EQUIVALENCE (THSTS(1),THS(1)),(THSTP(1),THP(1)),(TIS(1),TS(1)),(TI
     AP(1),TP(1)),(TIR(1),TR(1)),(DELRUD(1),RUD(1))
      FX = 0.0
      FY = 0.0
      FZ = 0.0
      FK = 0.0
      FM = 0.0
      FN = 0.0
      TL=TIME
      IF(NPR.EQ.0.0) GO TO 5
      RUDANG=FG1(TL,NPR,TR,RUD,IR)
      RUDANG=RUDANG/RAD
C    CALCULATE THRUSTS AND MOMENTS INDIVIDUALLY
      GO TO 6
    5 RUDANG=DELRUD(1)
      RUDANG=RUDANG/RAD
    6 CD=COS(RUDANG)
      SD=SIN(RUDANG)
      IF(NPS.EQ.0.0) GO TO 2
      THSS=FG1(TL,NPS,TS,THS,IS)
      GC TC 4
    2 THSS=THSTS(1)
    4 IF(NPP.EQ.0.0) GO TO 3
      THSP=FG1(TL,NPP,TP,THP,IP)
      GO TO 1
    3 THSP=THSTP(1)
    1 STHSTS=STHS*THSS
      STHSTP=STHP*THSP
      FXS=THSS*CD-STHSTS*SD
      FXP=THSP*CD+STHSTP*SD
      FYS=-STHSTS*CD-THSS*SD
      FYP=STHSTP*CD+SD*THSP
```

```
      FZS=-THSS*THETA*CD+STHSTS*SD*PHI
      FZP=-THSP*THETA*CD-STHSTP*SD*PHI
      FX=FXP+FXS
      FY=FYP+FYS
      FZ=FZP+FZS
      FKP=-FZP*YP-FYP*ZP
      FKS=FZS*YP-FYS*ZP
      FK=FKS+FKP
      FMS=FZS*(-XP)+FXS*ZP
      FMP=FZP*(-XP)+FXP*ZP
      FM=FMS+FMP
      FNS=-FXS*YP-FYS*(-XP)
      FNP=FXP*YP-FYP*(-XP)
      FN=FNS+FNP
      IF (IPROP.NE.ON) RETURN
                    WRITE(6,123)
     1FX,FY,FZ,FK,FM,FN
  123 FORMAT(/10X,22HPROP FX,FY,FZ,FK,FM,FN   /6E15.4)
      RETURN
      END


      SUBROUTINE RHS(VALUE)
      INTEGER ON
      COMMON /AIR/ PINF,RHOINF,GAM
      COMMON /BMCO / IMM,IMNX,IMNY,IBMFIL,BTIME,IMT,XMI(10),YMI(7),IX,IY
      COMMON /COLUMN/ IVERT,ILATRL
      COMMON /CONST/ PI,RAD,UO
      COMMON /CNTRL/CONTW,CONTQ,CONTTH,CMULT,LOUVER,ACONTZ,ACONTW,ZEQUIL
     1 ,THEQL,ACBASE
      COMMON/ENGINE/NPS,NPP,THSTS(25),THSTP(25),XP,YP,ZP,STHS,STHP,
     ATIP(25),TIS(25)
      COMMON /FANMAP/QIN,QBFAN(25),QMFAN(25),QSFAN(25),ENBFAN,ENMFAN,
     1 ENSFAN,BRPM,MRPM,SRPM,NPTSB,NPTSM,NPTSS
     2 ,PBFAN(25),PMFAN(25),PSFAN(25),TMEB(25),DELB(25),NB,TMES(25),
     3DETS(25),NS
      COMMON /FAERO/ FXAED,FYAED,FZAED,FKAED,FMAED,FNAED
      COMMON /FORBS/FXBS,FYBS,FZBS,FKBS,FMBS,FNBS,QLBS
      COMMON /FORSS/FXSS,FYSS,FZSS,FKSS,FMSS,FNSS,QLSS,FMS
      COMMON /FPROP/ FXP,FYP,FZP,FKP,FMP,FNP
      COMMON / FROUDE / FN,FNCRIT
      COMMON /FRUD/ FXRUD,FYRUD,FZRUD,FKRUD,FMRUD,FNRUD
      COMMON/GBOW/ XBOW
      COMMON /GEOM/     WIDTH,XL,XX(4,11),YY(4,11),NSTA(4),AB,VOLNOM
     1,DELS(4,10),XCPO,ZCP
      COMMON /GEOMBS/DETABX(11),DETABT(11),ARM1B(10),ARM2B(10)
     1 ,DFBS(10),TSKIB(10)
      COMMON /GEOMSS/DETADX(11),DETADT(11),ARM1S(10),DFSS(10),TSKIS(10)
```

```
1 ,ARM2S(10)                                                             RHS 0270
 COMMON /KSWTCH/ ITHRST                                                  RHS 0280
 COMMON /MASSES/ AM,AIXX,AIYY,AIZZ,AIXZ,AIMAX,G,WEIGHT,RHO,NMASS,       RHS 0290
-                AMI(201),XI(201),YI(201),ZI(201),XS,ZS,HRHO             RHS 0300
 COMMON /MATRIX/ A(6,6)                                                  RHS 0310
 COMMON /MSIDW/ DF(2,10),DSWAV(2,10),FXH(2),FYH(2),FZH(2),FMH(2),       RHS 0320
1               FNH(2),VFY(2),VFZ(2),FXV                                 RHS 0330
 COMMON /MWAVE/ FXW(2),FYW(2),FZW(2),FKW(2),FMW(2),FNW(2)               RHS 0340
 COMMON /OPTION/ I3DOF,ISRGE,ITRIM,NVALID                                RHS 0350
 COMMON /PLENUM/XLBW,XBBW,ABW,BUBHGT                                     RHS 0360
 COMMON /PRIME/ STIME,FTIME,DELT,DELPNT,TPRINT                           RHS 0370
 COMMON /PRTINT/ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,       RHS 0380
-IRUD,IPROP,IAEROD,IRHS                                                  RHS 0390
 COMMON /PWAVE/ FNCON,PWVCON                                             RHS 0400
 COMMON/ RUDDR/ NPR,DELRUD(25),XR,YR,ZR,IRDS,TL,RSPAN,RAREA,RASPR,      RHS 0410
ARCLB,RTC,RUDANG,TIR(25)                                                 RHS 0420
 COMMON /SIDE/FXSW,FYSW,FZSW,FKSW,FMSW,FNSW,ALSW,YSW,XLSW,CFSW,CDSWRHS 0430
1,VAREA   ,VCHORD,VSPAN,VANGLE,VCOS,VX,VY,VZ  ,AVBMSW,DELX,VTC           RHS 0440
 COMMON /SLOPE/WATSLP,XPWV,XLXPWV,XPWVXS                                 RHS 0450
 COMMON/SOFTBS/XBF,PBS,SINBS,COSBS,XBS,ZBS,DELYBS,DPBS,ELMAXB,YAVG RHS 0460
1B(10)                                                                   RHS 0470
 COMMON /SOFTSS/ XLF,PSS,SINTH,COSTH,XSS,ZSS,DELYSS,DPSS                 RHS 0480
1 ,ELMAXS,YAVGS(10)                                                      RHS 0490
 COMMON / VALOLD / YOLD(20)                                              RHS 0500
 COMMON /VARBLE/ VAL(40)                                                 RHS 0510
 COMMON /WAVE/ ETA(4,11),AW(10),OMEGA(10),DVOLW,NWAVE,BETA,             RHS 0520
1               FXWAV,FYWAV,FZWAV,FKWAV,FMWAV,FNWAV                      RHS 0530
2               ,ZBAR,PHIBAR,THEBAR,TC,COSBET,SINBET,PBBAR               RHS 0540
 EQUIVALENCE (VAL(1),TIME),(VAL(2),U),(VAL(3),V),(VAL(4),W),            RHS 0550
1(VAL(5),P),(VAL(6),Q),(VAL(7),R),(VAL(8),PHI),(VAL(9),THETA),          RHS 0560
2(VAL(10),Z),(VAL(11),BMASS),(VAL(21),X),(VAL(22),Y),(VAL(23),PSI),RHS 0570
3(VAL(24),PB)                                                            RHS 0580
 EQUIVALENCE (VAL(18),FANPWR)                                            RHS 0590
 DATA NOTIN /0/                                                          RHS 0600
 DIMENSION ACCEL(3),ANGACL(3)                                            RHS 0610
 DIMENSION GF(6),VALUE(20)                                               RHS 0620
 DO 5 J=1,20                                                             RHS 0630
5 VALUE(J)=0.0                                                           RHS 0640
C CALCULATION OF BUBBLE WAVE MAKING DRAG                                  RHS 0650
 FN=U/FNCON                                                              RHS 0660
 CF=.37/(FN**1.5655981)                                                  RHS 0670
 FXPWAV=-PWVCON*PBBAR*CF                                                 RHS 0680
 WATSLP=-FXPWAV/WEIGHT                                                   RHS 0690
 AB=ABW-(ABW-(XL*WIDTH))*(ZS+Z)/BUBHGT                                   RHS 0700
 VCL=VOLNOM-.5*(AB+ABW)*(Z+ZS)-DVOLW                                     RHS 0710
1+.5*WATSLP*XL*AB                                                        RHS 0720
 PB=PINF*(BMASS/(VOL*RHOINF))**GAM                                       RHS 0730
 PBS=PB+DPBS                                                             RHS 0740
```

```
      PSS=PB+DPSS                                                       RHS 0750
      PBAR=PB-PINF                                                      RHS 0760
      ABPB=PBAR*AB                                                      RHS 0770
      FLOW=SQRT(2.*ABS(PBAR)/RHOINF)*SIGN(1.,PBAR)                      RHS 0780
      QLSW=CFSW*ALSW*FLOW                                               RHS 0790
      CALL BOWSL                                                        RHS 0800
      CALL STNSL                                                        RHS 0810
                                                                        RHS 0820
      GF(1)=FXBS+FXSS+FXSW+FXRUD+FXP+FXWAV+FXAED            +FXPWAV     RHS 0830
      IF (ITHRST.NE.ITRIM) GO TO 11                                     RHS 0840
      THSTS(1)=THSTS(1)-GF(1)/2.                                        RHS 0850
      THSTP(1)=THSTP(1)-GF(1)/2.                                        RHS 0860
      THST=THSTS(1)+THSTP(1)                                            RHS 0870
      GF(1) = 0.0                                                       RHS 0880
   11 CONTINUE                                                          RHS 0890
      GF(2)=-R*U*AM+FYBS+FYSS+FYSW+FYRUD+FYP+FYWAV+FYAED                RHS 0900
      GF(3)=WEIGHT-ABPB        +FZBS+FZSS+FZSW+FZRUD+FZP+FZWAV+FZAED    RHS 0910
      GF(4)=FKBS+FKSS+FKSW+FKRUD+FKP+FKWAV+FKAED +ABPB*PHI*(-Z)         RHS 0920
      XCP=XCPO                                                          RHS 0930
      GF(5)=FMBS+FMSS+FMSW+FMRUD+FMP+FMWAV+FMAED +ABPB*(XCP-THETA*Z )   RHS 0940
     -          +FXPWAV*ZS                                              RHS 0950
      FMBUB=ABPB*(XCP-THETA*Z)                                          RHS 0960
      FWAVZ=FXPWAV*ZS                                                   RHS 0970
      GF(6)=FNBS+FNSS+FNSW+FNRUD+FNP+FNWAV+FNAED                        RHS 0980
      IF( I3DOF .NE. 1) GO TO 100                                       RHS 0990
      GF(3)=0.0                                                         RHS 1000
      GF(5)=0.0                                                         RHS 1010
  100 CONTINUE                                                          RHS 1020
      DO 1 I=1,6                                                        RHS 1030
      VALUE(I)=0.0                                                      RHS 1040
      DO 1 J=1,6                                                        RHS 1050
      VALUE(I)=VALUE(I)+A(I,J)*GF(J)                                    RHS 1060
    1 CONTINUE                                                          RHS 1070
      VALUE(7)=P                                                        RHS 1080
      VALUE(8)=Q                                                        RHS 1090
      VALUE(9)=W                                                        RHS 1100
      IF (I3DOF .EQ. 1 ) GO TO 325                                      RHS 1110
C     BUBBLE PRESSURE EQUATION                                          RHS 1120
      QOUT=QLBS+QLSS+QLSW                                               RHS 1130
      CALL FAN                                                          RHS 1140
      QCNTRL=0.0                                                        RHS 1150
      VALUE(10)=RHOINF*(QIN-QOUT-QCNTRL)                                RHS 1160
      GO TO 236                                                         RHS 1170
  325 CONTINUE                                                          RHS 1180
      VALUE(10)=0.0                                                     RHS 1190
  236 CONTINUE                                                          RHS 1200
C     WRITE DATA FILE FOR MOMENT AND SHEAR CALCS., IF REQUIRED          RHS 1210
      IF ( IMT.NE.1 )  GO TO 111                                        RHS 1220
```

```
      NBS = NSTA(3)-1
      NSS = NSTA(4)-1
      NSSL = NSS/2+1
                           WRITE (IBMFIL)  (VAL(I),I=1,24),ZBAR,PHIBAR,THEBAR,
     -                     FXW.FYW,FZW,FKW,FMW,FNW,  (VALUE(I),I=1,10),
     -                     DF,DSWAV,FXH,FYH,FZH,FMH,FNH,VFY,VFZ,FXV ,
     -                     FXRUD,FYRUD,FXP,FYP,FZP,   FZSS,FKSS,FMSS,FXBS,FZBS,
     -                     FXAED,FYAED,FZAED,FMAED,FNAED,FXPWAV,FXSS,FKBS,FMBS
     X ,FNBS,FNSS ,(TSKIS(I),DFSS(I),I=NSSL,NSS)
     Y            ,(TSKIB(I),DFBS(I),ARM1B(I),ARM2B(I),I=1,NBS)
  111 CONTINUE
C     CONSTANT LONGITUDINAL VELOCITY ( U )
      IF (ISRGE .EQ.1) VALUE(1)=0.0
      IF (ON.NE.1) RETURN
      DO 2 I=1,3
      ACCEL(I)=VALUE(I)/G
      ANGACL(I)=VALUE(I+3)*RAD
    2 CONTINUE
      BOWACC=ACCEL(3)-XBOW*VALUE(5)/G
      STNACC=ACCEL(3)+XS*VALUE(5)/G
      IF(IVERT.NE.ON) GO TO 10
      ZD=Z+ZS
      THETAR=THETA*RAD
   10 IF(ILATRL.NE.ON) GO TO 15
      DEPSI=PSI*RAD
      RDEG=R*RAD
      BETAS=-V/U*RAD
      ACCLAT=(VALUE(2)+U*R)/G
      DPHI=PHI*RAD
      DRFT=12.0*ZD
      VEL = 0.59249*U
      DELRS=RUDANG*RAD
      IF(R.EQ.0.0) GO TO 115
      TRADUS=U/R
      GO TO 20
  115 TRADUS=1.E8
   20 WRITE(1)TIME,VAL(16),DRFT,THETAR,PBAR,BOWACC,ACCEL(3),FANPWR,DPHI,
     1DEPSI,ACCLAT,VEL,TRADUS,RDEG,X,Y,QIN,QOUT,GF(1),FXPWAV,GF(2),GF(3)
     2,GF(4),GF(5),GF(6),DELRS
   15 IF (IRHS .NE. ON) RETURN
       WRITE(6,77) FMBS,FMSS,FMSW,FMRUD,FMP,FMWAV,FMAED,FMBUB,FWAVZ
   77 FORMAT('0',6X,5HFMBS=,E16.6,2X,5HFMSS=,E16.6,2X,5HFMSW=,
     AE16.6,/,'0',6X,6HFMRUD=,E16.6,2X,4HFMP=,E16.6,2X,6HFMWAV=,E16.6,
     B/,'0',6X,6HFMAED=,E16.6,2X,6HFMBUB=,E16.6,2X,6HFWAVZ=,E16.6)
      WRITE(6,401)FXPWAV
  401 FORMAT('0',6X,7HFXPWAV=,E16.6)
      WRITE(6,200) PBAR,FANPWR,QIN,QLBS,QLSW,QLSS
      WRITE(6,215) AB,VOL
```

```
                 WRITE(6,213) VALUE,VAL                                    RHS 1710
      WRITE(6,150) GF,ACCEL,ANGACL                                         RHS 1720
      WRITE(6,175)BOWACC,STNACC                                            RHS 1750
  200 FORMAT(//10X,3HRHS                                                   RHS 1760
     120H GAGE PRESS. (PSF) = F7.2,5X,21HFAN POWER REQD (HP) =  F8.2,     RHS 1770
     25X,27HFAN FLOW RATE (CU FT/SEC) =  F9.2,                             RHS 1780
     3//31H LEAKAGE FLOW RATES (CU FT/SEC)  //11H BOW SEAL = F9.2,         RHS 1790
     411H SIDEWALL = F9.2,13H STERN SEAL = F9.2)                           RHS 1800
  215 FORMAT(/13H PLENUM AREA= F9.2,10X,14HPLENUM VOLUME= F10.2)          RHS 1810
  213 FORMAT(//12H VALUE ARRAY 2(/10E13.4)/10H VAL ARRAY 4(/10E13.4))     RHS 1820
  150 FORMAT(/10X,24HTOTAL FORCES AND MOMENTS 6E12.4/10X,24HACCELERATIONRHS 1830
     -S G,DEG/SEC2 6E12.4)                                                 RHS 1840
  175 FORMAT(/10X,16HBOW ACCEL. (G) = E12.4,21H   STERN ACCEL. (G) = E12RHS 1850
     -.4)                                                                  RHS 1860
      RETURN                                                               RHS 1870
      END                                                                  RHS 1880


      SUBROUTINE RUDDER                                                    RUD 0020
      INTEGER ON                                                           RUD 0030
      COMMON /CONST/ PI,RAD,U0                                             RUD 0040
      COMMON /FRUD/ FX,FY,FZ,FK,FM,FN                                      RUD 0050
      COMMON /MASSES/ AM,AIXX,AIYY,AIZZ,AIXZ,AIMAX,G,WEIGHT,RHO,NMASS,    RUD 0060
     -                AMI(201),XI(201),YI(201),ZI(201),XS,ZS,HRHO          RUD 0070
      COMMON /PRTINT/ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,    RUD 0080
     -IRUD,IPROP,IAEROD,IRHS                                               RUD 0090
      COMMON/RUDDR/ NPR,DELRUD(25),XR,YR,ZR,IRDS,TL,RSPAN,RAREA,RASPR,    RUD 0100
     ARCLB,RTC,RUDANG,TIR(25)                                              RUD 0110
      COMMON /VARBLE/ VAL(40)                                              RUD 0120
                                                                           RUD 0130
      EQUIVALENCE (VAL(1),TIME),(VAL(2),U),(VAL(3),V),(VAL(4),W),         RUD 0140
     1(VAL(5),P),(VAL(6),Q),(VAL(7),R),(VAL(8),PHI),(VAL(9),THETA),       RUD 0150
     2(VAL(10),Z),(VAL(11),BMASS),(VAL(21),X),(VAL(22),Y),(VAL(23),PSI),RUD 0160
     3(VAL(24),PB)                                                         RUD 0170
      EQUIVALENCE (DELRUD(1),RUD(1)),(TIR(1),TR(1))                        RUD 0180
      DIMENSION RUD(1),TR(1)                                               RUD 0190
      EQUIVALENCE (VAL(18),FANPWR)                                         RUD 0200
      DATA ENU /1.28E-5/                                                   RUD 0210
                                                                           RUD 0220
C     CALCULATE PROGRAMMED RUDDER DEFLECTION                               RUD 0230
      TL=TIME                                                              RUD 0240
      IF(NPR.EQ.0.0) GO TO 5                                               RUD 0250
      GO TO 6                                                              RUD 0260
    5 RUDANG=DELRUD(1)                                                     RUD 0270
      RUDANG=RUDANG/RAD                                                    RUD 0280
      GO TO 7                                                              RUD 0290
    6 RUDANG=FG1(TL,NPR,TR,RUD,IR)                                         RUD 0300
      RUDANG=RUDANG/RAD                                                    RUD 0310
```

```
C     SIDE FORCE ON RUDDER                                               RUD 0320
    7 DSR=Z+ZS-XR*THETA                                                  RUD 0330
      ENDFAC=(1.+DSR/(DSR+RSPAN))                                        RUD 0340
      VH=V+XR*R-ZR*P                                                     RUD 0350
      QQ=HRHO*U*U*RAREA                                                  RUD 0360
      EFFANG=RUDANG-ENDFAC*VH/U                                          RUD 0370
      FY=2.*QQ*ENDFAC*RCLB*EFFANG                                        RUD 0380
C     DRAG FORCE ON RUDDER                                               RUD 0400
      REY=U*(RAREA/RSPAN)/ENU                                            RUD 0410
      CFR=.42//(ALOG10(REY)-.407)**2.64                                  RUD 0420
      PI8=PI/8.                                                          RUD 0430
      CD=2.*CFR+ PI8*RTC*RTC*(1.+G*RSPAN/(U*U))+RCLB*EFFANG*EFFANG       RUD 0440
      FX=-2.*CD*RAREA*HRHO*U*U                                           RUD 0450
      FZ=0.                                                              RUD 0470
      FK=-ZR*FY                                                          RUD 0480
      FM=FX*ZR                                                           RUD 0490
      FN=XR*FY                                                           RUD 0500
      IF(IRUD.NE.ON) RETURN                                              RUD 0510
                            WRITE(6,123)                                 RUD 0520
     1FX,FY,FZ,FK,FM,FN                                                  RUD 0530
  123 FORMAT(/10X,24HRUDDER FX,FY,FZ,FK,FM,FN /6E15.4)                   RUD 0540
      RETURN                                                             RUD 0550
      END                                                                RUD 0560

      SUBROUTINE SAM                                                     SAM 0020
      WRITE (6,10)                                                       SAM 0030
   10 FORMAT (1H1,'YOU HAVE CALLED A DUMMY SAM SUBROUTINE.'/             SAM 0040
     110X,'CHANGE TO BHISES TO USE THE SAM SUBROUTINE.')                 SAM 0050
      RETURN                                                             SAM 0060
      END                                                                SAM 0070

C                                                                        T1  0010
      FUNCTION T1(X)                                                     T1  0020
      IF(ABS(X)-.1) 10,10,20                                             T1  0030
   10 T1=X*(1.-X*X/10.0)/3.                                              T1  0040
      RETURN                                                             T1  0050
   20 T1=(SIN(X)-X*COS(X))/(X*X)                                         T1  0060
      RETURN                                                             T1  0070
      END                                                                T1  0080

C                                                                        T2  0010
      FUNCTION T2(X)                                                     T2  0020
      IF(ABS(X)-.1) 10,10,20                                             T2  0030
   10 T2=1.-X*X/6.                                                       T2  0040
      RETURN                                                             T2  0050
```

```
20    T2=SIN(X)/X                                                     T2  0060
      RETURN                                                          T2  0070
      END                                                             T2  0080

      SUBROUTINE SIDEWL                                               SDWL0010
      INTEGER ON                                                      SDWL0020
      COMMON /AIR/ PINF,RHOINF,GAM                                    SDWL0030
      COMMON /BMCO / IMM,IMNX,IMNY,IBMFIL,BTIME,IMT,XMI(10),YMI(7),IX,IYSDWL0040
      COMMON /CONST/ PI,RAD,UO                                        SDWL0050
      COMMON /GEOM/    WIDTH,XL,XX(4,11),YY(4,11),NSTA(4),AB,VOLNCM   SDWL0060
     1,DELS(4,10),XCP,ZCP                                             SDWL0070
      COMMON /GEOMSW/ XAVG(10),DS                                     SDWL0080
      COMMON /KSWTCH/ ITHRST                                          SDWL0090
      COMMON /MASSES/ AM,AIXX,AIYY,AIZZ,AIXZ,AIMAX,G,WEIGHT,RHO,NMASS, SDWL0100
     -                AMI(201),XI(201),YI(201),ZI(201),XS,ZS,HRHO     SDWL0110
      COMMON /MSIDW/ DF(2,10),DSWAV(2,10),FXH(2),FYH(2),FZH(2),FMH(2), SDWL0120
     1               FNH(2),VFY(2),VFZ(2),FXV                         SDWL0130
      COMMON /PLENUM/XLBW,XBBW,ABW,BUBHGT                             SDWL0140
      COMMON /PRIME/ STIME,FTIME,DELT,DELPNT,TPRINT                   SDWL0150
      COMMON /PRTINT/ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES, SDWL0160
     -IRUD,IPROP,IAEROD,IRHS                                          SDWL0170
      COMMON /SIDE/FX   ,FY   ,FZ   ,FK   ,FM   ,FN   ,ALSW,YSW,XLSW,CFSW,CDSWSDWL0180
     1,VAREA  ,VCHORD,VSPAN,VANGLE,VCOS,VX,VY,VZ  ,AVBMSW,DELX,VTC    SDWL0190
      COMMON /SLOPE/WATSLP,XPWV,XLXPWV,XPWVXS                         SDWL0200
      COMMON /VARBLE/ VAL(40)                                         SDWL0210
      COMMON /WAVE/ ETA(4,11),AW(10),OMEGA(10),DVOLW,NWAVE,BETA,      SDWL0220
     1              FXWAV,FYWAV,FZWAV,FKWAV,FMWAV,FNWAV               SDWL0230
     2              ,ZBAR,PHIBAR,THEBAR,TC,COSBET,SINBET              SDWL0240
      COMMON /WAVTAB/ NAL,DAL,SAL,NDS,DDS,SDS,NTH,DTH,STH,NBB,DBB,SBB, SDWL0250
     1         AC1(20,5,7),AC2(20,5,7),AC3(20,5,7),AC4(20,5,7),       SDWL0260
     2         AC5(20,5,7),AC6(20,5,7),AC7(20,5,7)                    SDWL0270
     3         ,AC0(20,5,7),AC00(20,5,7),AC8(20,5,7),                 SDWL0280
     4         AS1(20,5,7),AS2(20,5,7),AS3(20,5,7),AS4(20,5,7),       SDWL0290
     5         AS5(20,5,7),AS6(20,5,7),AS7(20,5,7)                    SDWL0300
     6         ,AS0(20,5,7),AS00(20,5,7),AS8(20,5,7)                  SDWL0310
     7         ,BB(36),XREF,KX                                        SDWL0320
      EQUIVALENCE           (VAL(2),U),(VAL(3),V),(VAL(4),W),         SDWL0330
     1(VAL(5),P),(VAL(6),Q),(VAL(7),R),(VAL(8),PHI),(VAL(9),THETA),   SDWL0340
     2(VAL(10),Z),(VAL(11),BMASS),(VAL(21),X),(VAL(22),Y),(VAL(23),PSI),SDWL0350
     3(VAL(24),PB)                                                    SDWL0360
      DIMENSION GAP(2,11),DSW(2,11)                                   SDWL0370
      DATA ENU /1.28E-5/                                              SDWL0380
                                                                      SDWL0390
      PBHEAD=PBAR/(RHO*G)                                             SDWL0400
C     GAP OR WETTED DRAFT CALCULATION                                 SDWL0410
      DO 10 J=1,2                                                     SDWL0420
      N=NSTA(J)                                                       SDWL0430
```

```
      DO 10 K=1,N
      DD=ZS+Z+YY(J,K)*PHI-XX(1,K)*THETA+ETA(J,K)
      DDIN=DD-WATSLP*(XPWVXS-XX(J,K))
      IF(DDIN.LT.BUBHGT) GO TO 101
      IF ( VAL(1)-TOLD .LT. DELPNT )  GO TO 101
      TOLD = VAL(1)
      WRITE (6,100) XX(J,K),VAL(1),DD
  100 FORMAT(/10X,43HWATER CONTACT WITH TOP OF BUBBLE CHAMBER AT F7.2,
     -14H FT.     TIME = F7.2,19H SEC.     IMMERSION= F7.2,4H FT. )
  101 CONTINUE
      DSW(J,K)=(SIGN(1.,DD)+1.)*DD/2.
      IF (DDIN) 6,8,8
    6 IF (DSW(J,K)-PBHEAD) 7,8,8
    7 GAP(J,K)=-DDIN*(1.-(DSW(J,K))/PBHEAD)
      GO TO 10
    8 GAP(J,K)=0.0
   10 CONTINUE
C     LEAKAGE AREA
      ALSW=0.0
      DO 20 J=1,2
      N=NSTA(J)-1
      DO 20 I=1,N
      ALSW=ALSW+(GAP(J,I)+GAP(J,I+1))*DELX/2.
   20 CONTINUE

C     CROSS-FLOW DRAG ON SIDEWALLS
      FYD=0.0
      FKD=0.0
      FND=0.0
      DO 15 I=1,2
      N=NSTA(I)-1
      DO 15 J=1,N
      DSWAV(I,J)=(DSW(I,J)+DSW(I,J+1))/2.
      VREL     =V+XAVG(J)*R-(ZS-DSWAV(I,J)/2.)*P
      DF(I,J)=-  HRHO*CDSW*VREL     *ABS(VREL     )*DELX     *DSWAV(I,J)
      FYD=FYD+DF(I,J)
      FND=FND+DF(I,J)*XAVG(J)
   15 FKD=FKD-(ZS-DSWAV(I,J)/2.)*DF(I,J)
C     SET UP STERN LIMIT OF FORCE DETERMINATION
      XSS = -XS
      GO TO 16
      ENTRY SIDWLM
      XSS = XMI(IX)
   16 IP=1.+(THETA*RAD-STH)/DTH
      IP=MAX0(MIN0(IP,NTH),1)
      IP1=MIN0(IP+1,NTH)
      DTHETA=(IP-1)*DTH+STH
      DIP= (THETA*RAD-DTHETA)/DTH
```

```
C     CALC REYNOLDS NO. AND DRAG COEFF.                                 SDWL0920
      REY=U*XLSW/ENU                                                    SDWL0930
      CDT=.427/(ALOG10(REY)-.407)**2.64                                 SDWL0940
                                                                        SDWL0950
C     SIDEWALL FORCES, P/S                                              SDWL0960
      DO 40 J=1,2                                                       SDWL0970
      WAREA=0.0                                                         SDWL0980
      N=NSTA(J)-1                                                       SDWL0990
      NI = (XSS+XS)*N/XLSW+1.5                                          SDWL1000
      DO 21 I=NI,N                                                      SDWL1010
      ZOR1=1.                                                           SDWL1020
      IF(DSWAV(J,I).EQ. 0.0) ZOR1=0.0                                   SDWL1030
   21 WAREA=WAREA+DELX*(2.*DSWAV(J,I)+ZOR1*AVBMSW)                      SDWL1040
      FXH(J)=-  HRHO*CDT*WAREA*U*U                                      SDWL1050
      PM1=2*J-3                                                         SDWL1060
      YLSW=PM1*YSW                                                      SDWL1070
      DS= Z   +ZS+YLSW*PHI                                              SDWL1080
      DSS=DS-XSS*THETA                                                  SDWL1090
      ZOR1=(SIGN(1.,DSS)+1.)/2.                                         SDWL1100
      DSS=DSS*ZOR1                                                      SDWL1110
      IDSS=1.5+(DSS-SBB)/DBB                                            SDWL1120
      IDSS=MIN0(NBB,IDSS)                                               SDWL1130
      BS=BB(IDSS)                                                       SDWL1140
      ZOR1=(SIGN(1.,DSS)+1.)/2.                                         SDWL1150
      DSS=DSS*ZOR1                                                      SDWL1160
      A33S=(RHO*PI*BS**2)/8.                                            SDWL1170
      A22S=(RHO*.4*PI*DSS**2)/2.                                        SDWL1180
      DSR=DS-(XREF-XS)*THETA                                            SDWL1190
      ID=1.+(DSR*12.-SDS)/DDS                                           SDWL1200
      ID=MAX0(MIN0(ID,NDS),1)                                           SDWL1210
      DDSR =(ID-1)*DDS  +SDS                                            SDWL1220
      ID1=MIN0(ID+1,NDS)                                                SDWL1230
      DID= (DSR*12.-DDSR)/DDS                                           SDWL1240
      BCO=ACO( 1,ID,IP)                                                 SDWL1250
      BCOO=ACOO( 1,ID,IP)                                               SDWL1260
      BC2=AC2( 1,ID,IP)                                                 SDWL1270
      BC5 =  AC5(1,ID,IP)                                               SDWL1280
      BC6 =  AC6(1,ID,IP)                                               SDWL1290
      BCO =BCO  +DID*(ACO  (1,ID1,IP)-BCO  ) +DIP*(ACO  (1,ID,IP1)-BCO  SDWL1300
     1   +DID*(ACO  (1,ID1,IP1)-ACO  (1,ID,IP1)-ACO  (1,ID1,IP)+BCO  )) SDWL1310
      BCOO=BCOO+DID*(ACOO(1,ID1,IP)-BCOO) +DIP*(ACOO(1,ID,IP1)-BCOO     SDWL1320
     1   +DID*(ACOO(1,ID1,IP1)-ACOO(1,ID,IP1)-ACOO(1,ID1,IP)+BCOO))     SDWL1330
      BC2 =BC2  +DID*(AC2  (1,ID1,IP)-BC2  ) +DIP*(AC2  (1,ID,IP1)-BC2  SDWL1340
     1   +DID*(AC2  (1,ID1,IP1)-AC2  (1,ID,IP1)-AC2  (1,ID1,IP)+BC2  )) SDWL1350
      BC5 =BC5  +DID*(AC5  (1,ID1,IP)-BC5  ) +DIP*(AC5  (1,ID,IP1)-BC5  SDWL1360
     1   +DID*(AC5  (1,ID1,IP1)-AC5  (1,ID,IP1)-AC5  (1,ID1,IP)+BC5  )) SDWL1370
      BC6 =BC6  +DID*(AC6  (1,ID1,IP)-BC6  ) +DIP*(AC6  (1,ID,IP1)-BC6  SDWL1380
     1   +DID*(AC6  (1,ID1,IP1)-AC6  (1,ID,IP1)-AC6  (1,ID1,IP)+BC6  )) SDWL1390
```

```
C     SHIFT MOMENT CENTER FROM XREF TO C.G.                             SDWL1400
      BC00 = BC00-(XS-XREF)*BC0                                         SDWL1410
      BC6  = BC6 -(XS-XREF)*BC5                                         SDWL1420
C     HYDROSTATIC AND HYDRODYNAMIC FORCES                               SDWL1430
      FZH(J) =-G*BC0-U*U*A33S*THETA-U*A33S*W+Q*U*(-BC2+A33S*XSS)        SDWL1440
     1   -U*A33S*P*YLSW                                                 SDWL1450
      FMH(J) =-U*XSS*XSS*A33S*Q+G*BC00+U*(A33S*XSS+BC2)*(W+U*THETA      SDWL1460
     1  +YLSW*P)                                                        SDWL1470
      FYH(J) =-A22S*U*(V+XSS*R  -ZS*P)                                  SDWL1480
      FNH(J) = FYH(J)*XSS-U*((V-ZS*P)*BC5+R*BC6)                        SDWL1490
      IF (IMT.EQ.2)  GO TO 40                                           SDWL1500
39    CONTINUE                                                          SDWL1510
40    CONTINUE                                                          SDWL1520
      IF (IMT.EQ.2)  RETURN                                             SDWL1530
C     TOTAL SIDEWALL FORCES AMD MOMENTS                                 SDWL1540
      FX=FXH(1)+FXH(2)                                                  SDWL1550
      FY=FYH(1)+FYH(2)                                                  SDWL1560
      FZ=FZH(1)+FZH(2)                                                  SDWL1570
      FK=(FZH(2)-FZH(1))*YSW          +FKD -FY*ZS                       SDWL1580
      FY=FY+FYD                                                         SDWL1590
      FM=FMH(1)+FMH(2)+ZS*FX                                            SDWL1600
      FN= FND +FNH(1)+FNH(2) +(FXH(1)-FXH(2))*YSW                       SDWL1610
      IF (ISIDWL.NE.ON) RETURN                                          SDWL1620
      DO 41 I=1,2                                                       SDWL1630
      DO 41 J=1,11                                                      SDWL1640
      GAP(I,J)=12.0*GAP(I,J)                                            SDWL1650
   41 DSW(I,J)=12.0*DSW(I,J)                                            SDWL1660
      WRITE( 6,123) ((GAP(I,J),J=1,11),I=1,2),((DSW(I,J),J=1,11),I=1,2),SDWL1670
     1 FX,FY,FZ,FK,FM,FN                                                SDWL1680
  123 FORMAT(/10X,8HSIDEWALL /25H GAP (FT.) (STERN TO BOW) /14H PORT SIDSDWL1690
     1EWALL /11F10.5/14H STBD SIDEWALL /11F10.5/37H IMMERSION DEPTH (FT.SDWL1700
     2) (STERN TO BOW) /14H PORT SIDEWALL /11F10.5/14H STBD SIDEWALL /  SDWL1710
     3 11F10.5/10X,26HSIDEWALL FX,FY,FZ,FK,FM,FN /6E15.4)               SDWL1720
      RETURN                                                            SDWL1730
      END                                                               SDWL1740


      SUBROUTINE STNSL                                                  SSL 0010
      INTEGER ON                                                        SSL 0020
      COMMON /AIR/ PINF,RHOINF,GAM                                      SSL 0030
      COMMON /CONST/ PI,RAD,U0                                          SSL 0040
      COMMON / FORSS/ FX,FY,FZ,FK,FM,FN,QL,FMS                          SSL 0050
      COMMON /GEOM/    WIDTH,XL,XX(4,11),YY(4,11),NSTA(4),AB,VOLNOM     SSL 0060
     1,DELS(4,10),XCP,ZCP                                               SSL 0070
      COMMON /GEOMSS/DETADX(11),DETADT(11),ARM1S(10),DFSS(10),TSKIS(10) SSL 0080
     1 ,ARM2S(10)                                                       SSL 0090
      COMMON /KSWTCH/ ITHRST                                            SSL 0100
      COMMON/LEAKER/ ALEAK,CFSS                                         SSL 0110
```

```
      COMMON /MASSES/  AM,AIXX,AIYY,AIZZ,AIXZ,AIMAX,G,WEIGHT,RHO,NMASS,   SSL 0120
     -                 AMI(201),XI(201),YI(201),ZI(201),XS,ZS,HRHO         SSL 0130
      COMMON /PRTINT/ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,     SSL 0140
     -IRUD,IPROP,IAEROD,IRHS                                               SSL 0150
      COMMON /SLOPE/WATSLP,XPWV,XLXPWV,XPWVXS                              SSL 0160
      COMMON /SOFTSS/ XLF,PSS,SINTH,COSTH,XSS,ZSS,DELYSS,DPSS              SSL 0170
     1 ,ELMAXS,YAVGS(10)                                                   SSL 0180
      COMMON /STSLR/ CPHI,CPHID                                            SSL 0190
      COMMON / VALOLD / YOLD(20)                                           SSL 0200
      COMMON /VARBLE/ VAL(40)                                              SSL 0210
      COMMON /WAVE/ ETA(4,11),AW(10),OMEGA(10),DVOLW,NWAVE,BETA,           SSL 0220
     1              FXWAV,FYWAV,FZWAV,FKWAV,FMWAV,FNWAV                    SSL 0230
     2              ,ZBAR,PHIBAR,THEBAR,TC,COSBET,SINBET,PBBAR             SSL 0240
      EQUIVALENCE (VAL(1),TIME),(VAL(2),U),(VAL(3),V),(VAL(4),W),          SSL 0250
     1(VAL(5),P),(VAL(6),Q),(VAL(7),R),(VAL(8),PHI),(VAL(9),THETA),        SSL 0260
     2(VAL(10),Z),(VAL(11),BMASS),(VAL(21),X),(VAL(22),Y),(VAL(23),PSI),SSL 0270
     3(VAL(24),PB)                                                         SSL 0280
      DIMENSION GAP(11),ELSKI(11)                                          SSL 0290
      DATA ENU,UWSKI,CLSKI  / 1.28E-5, 0.0,  1.5708/                       SSL 0300
      DO 5 J=1,11                                                          SSL 0310
      GAP(J)=0.0                                                           SSL 0320
      ELSKI(J)=0.0                                                         SSL 0330
    5 CONTINUE                                                             SSL 0340
      ALSS=0.0                                                             SSL 0350
      FX=0.0                                                               SSL 0360
      FZ=0.0                                                               SSL 0370
      FK=0.0                                                               SSL 0380
      FM=0.0                                                               SSL 0390
      FN=0.0                                                               SSL 0400
      DELP=PSS-PB                                                          SSL 0410
      IF (DELP.LT.0.0) DELP=0.0                                            SSL 0420
      PBAR=PB-PINF                                                         SSL 0430
C     CALCULATE ELSKI HERE.                                                SSL 0440
      SINDIF=SINTH-COSTH*THETA                                             SSL 0450
      COSDIF=COSTH+SINTH*THETA                                             SSL 0460
      X1=XSS+ZSS*THETA-XLF*SINDIF                                          SSL 0470
      Z1=(-Z-ZSS+XSS*THETA-XLF*COSDIF)                                     SSL 0480
C     CALCULATE GAP HERE.                                                  SSL 0490
      N=NSTA(4)                                                            SSL 0500
      DO 10 K=1,N                                                          SSL 0510
      ELSKI(K)=(ETA(4,K)-DETADX(K)*(XX(4,K)-X1)-Z1)+YY(4,K)*PHI            SSL 0520
     1-XPWV*WATSLP                                                         SSL 0530
      GAP(K)=-ELSKI(K)                                                     SSL 0540
      IF (GAP(K) .LT. 0.0) GAP(K)=0.0                                      SSL 0550
   10 CONTINUE                                                             SSL 0560
      N=NSTA(4)-1                                                          SSL 0570
      DO 20 J=1,N                                                          SSL 0580
      ELSKIA=(ELSKI(J+1)+ELSKI(J))/2.                                      SSL 0590
```

```
      IF (ELSKIA.LE.0.0) GO TO 15                                       SSL 0600
      IF (ELSKIA.GT.ELMAXS) ELSKIA=ELMAXS                               SSL 0610
      ARM1S(J)=XX(4,J)+ELSKIA/2.                                        SSL 0620
      ARM2S(J)=ZS-ELSKIA                                                SSL 0630
      DFSS(J)=-DELP*ELSKIA*DELYSS                                       SSL 0640
      ARG=.5*RHO*U*U*ELSKIA*DELYSS                                      SSL 0650
      RESKI =U*ELSKIA/ENU                                               SSL 0660
      CDTSKI=.427/(ALOG10(RESKI)-.407)**2.64                            SSL 0670
      TSKIS(J) = - ARG*CDTSKI                                           SSL 0680
      GO TO 18                                                          SSL 0690
   15 DFSS(J) = 0.0                                                     SSL 0700
      TSKIS(J) = 0.0                                                    SSL 0710
   18 CONTINUE                                                          SSL 0720
      FX=FX+TSKIS(J)                                                    SSL 0730
      FZ=FZ+DFSS(J)                                                     SSL 0740
      FK=FK+DFSS(J)*YAVGS(J)                                            SSL 0750
      FM=FM-DFSS(J)*ARM1S(J)+TSKIS(J)*ARM2S(J)                          SSL 0760
      FN=FN-TSKIS(J)*YAVGS(J)                                           SSL 0770
      ALSS=ALSS+(GAP(J)+GAP(J+1))*DELYSS/2.0                            SSL 0780
   20 CONTINUE                                                          SSL 0790
      ALSS=ALSS+ALEAK                                                   SSL 0800
      QL=CFSS*ALSS*SQRT(2.*ABS(PBAR)/RHOINF)*SIGN(1.,PBAR)              SSL 0810
      IF (ISTNSL.NE.ON) RETURN                                          SSL 0820
      WRITE(6,100) GAP,ELSKI,FX,FY,FZ,FK,FM,FN                          SSL 0830
  100 FORMAT(//12H  STERN SEAL /26H  GAP (FT.) PORT TO STBD.  /11E11.3  SSL 0840
     1/28H  ELSKI (FT.) PORT TO STBD.  /11E11.3/10X,23HSTNSL FX,FY,FZ,FKSSL 0850
     2,FM,FN /6E15.4)                                                   SSL 0860
      RETURN                                                            SSL 0870
      END                                                               SSL 0880

      SUBROUTINE WAVES(TIME)                                            WAVS0010
      INTEGER ON                                                        WAVS0020
      COMMON /BMCO / IMM,IMNX,IMNY,IBMFIL,BTIME,IMT,XMI(10),YMI(7),IX,IYWAVS0030
      COMMON /CONST/ PI,RAD,UO                                          WAVS0040
      COMMON /GEOM/    WIDTH,XL,XX(4,11),YY(4,11),NSTA(4),AB,VOLNOM     WAVS0050
     1,DELS(4,10),XCPO,ZCP                                              WAVS0060
      COMMON /GEOMBS/DETABX(11),DETABT(11),ARM1B(10),ARM2B(10)          WAVS0070
     1 ,DFBS(10),TSKIB(10)                                              WAVS0080
      COMMON /GEOMSS/DETADX(11),DETADT(11),ARM1S(10),DFSS(10),TSKIS(10) WAVS0090
     1 ,ARM2S(10)                                                       WAVS0100
      COMMON /MASSES/ AM,AIXX,AIYY,AIZZ,AIXZ,AIMAX,G,WEIGHT,RHO,NMASS,  WAVS0110
     -                AMI(201),XI(201),YI(201),ZI(201),XS,ZS,HRHO       WAVS0120
      COMMON /MWAVE/ FXW(2),FYW(2),FZW(2),FKW(2),FMW(2),FNW(2)          WAVS0130
      COMMON /PLENUM/XLBW,XBBW,ABW,BUBHGT                               WAVS0140
      COMMON /PRTINT/ON,IACCEL,IVEL,ITRAJ,ISIDWL,IBOWSL,ISTNSL,IWAVES,  WAVS0150
     -IRUD,IPROP,IAEROD,IRHS                                            WAVS0160
      COMMON /RISER/ AMPTC                                              WAVS0170
```

```
      COMMON /SIDE/FXSW,FYSW,FZSW,FKSW,FMSW,FNSW,ALSW,YSW,XLSW,CFSW,CDSWWAVS0180
     1,VAREA  ,VCHORD,VSPAN,VANGLE,VCOS,VX,VY,VZ  ,AVBMSW,DELX,VTC        WAVS0190
      COMMON /VARBLE/ VAL(40)                                             WAVS0200
      COMMON /WAVE/ ETA(4,11),AW(10),OMEGA(10),DVOLW,NWAVE,BETA,          WAVS0210
     1              FXWAV,FYWAV,FZWAV,FKWAV,FMWAV,FNWAV                   WAVS0220
     2              ,ZBAR,PHIBAR,THEBAR,TC,COSBET,SINBET,PBBAR            WAVS0230
      COMMON /WAVTAB/ NAL,DAL,SAL,NDS,DDS,SDS,NTH,DTH,STH,NBB,DBB,SBB,    WAVS0240
     1          AC1(20,5,7),AC2(20,5,7),AC3(20,5,7),AC4(20,5,7),          WAVS0250
     2          AC5(20,5,7),AC6(20,5,7),AC7(20,5,7)                       WAVS0260
     3          ,AC0(20,5,7),AC00(20,5,7),AC8(20,5,7),                    WAVS0270
     4          AS1(20,5,7),AS2(20,5,7),AS3(20,5,7),AS4(20,5,7),          WAVS0280
     5          AS5(20,5,7),AS6(20,5,7),AS7(20,5,7)                       WAVS0290
     6          ,AS0(20,5,7),AS00(20,5,7),AS8(20,5,7)                     WAVS0300
     7          ,BB(36),XREF,RX                                           WAVS0310
      DIMENSION WC0(2),WC00(2),WC1(2),WC2(2),WC3(2),WC4(2),WC5(2),WC6(2)WAVS0320
     1         ,WC7(2),WC8(2)                                             WAVS0330
      DIMENSION WS0(2),WS00(2),WS1(2),WS2(2),WS3(2),WS4(2),WS5(2),WS6(2)WAVS0340
     1         ,WS7(2),WS8(2)                                             WAVS0350
      EQUIVALENCE             (VAL(2),U),(VAL(3),V),(VAL(4),W),           WAVS0360
     1(VAL(5),P),(VAL(6),Q),(VAL(7),R),(VAL(8),PHI),(VAL(9),THETA),       WAVS0370
     2(VAL(10),Z),(VAL(11),BMASS),(VAL(21),X),(VAL(22),Y),(VAL(23),PSI),WAVS0380
     3(VAL(24),PB)                                                        WAVS0390
      EQUIVALENCE (VAL(16),ETACG)                                         WAVS0400
      IF (NWAVE.EQ.0) RETURN                                              WAVS0410
      XCP=XCPO                                                            WAVS0420
      GAMMA=BETA-PSI                                                      WAVS0430
      SIGAM=SIN(GAMMA)                                                    WAVS0440
      COGAM=COS(GAMMA)                                                    WAVS0450
      FO= -X*COSBET -Y*SINBET                                             WAVS0460
      DVOLW=0.0                                                           WAVS0470
      ETACG=0.0                                                           WAVS0480
      N=NSTA(3)                                                           WAVS0490
      DO 1 J=1,N                                                          WAVS0500
      DETABX(J)=0.0                                                       WAVS0510
    1 CONTINUE                                                            WAVS0520
      N=NSTA(4)                                                           WAVS0530
      DO 2 J=1,N                                                          WAVS0540
      DETADX(J)=0.0                                                       WAVS0550
    2 CONTINUE                                                            WAVS0560
      DO 10 J=1,4                                                         WAVS0570
      N=NSTA(J)                                                           WAVS0580
      DO 10 K=1,N                                                         WAVS0590
   10 ETA(J,K)=0.0                                                        WAVS0600
      DO 15 J=1,2                                                         WAVS0610
      FXW(J) = 0.0                                                        WAVS0620
      FYW(J) = 0.0                                                        WAVS0630
      FZW(J) = 0.0                                                        WAVS0640
      FKW(J) = 0.0                                                        WAVS0650
```

```
      FMW(J) = 0.0                                                      WAVS0660
      FNW(J) = 0.0                                                      WAVS0670
   15 CONTINUE                                                          WAVS0680
      XSS = -XS                                                         WAVS0690
      IF (IMT.EQ.2)  XSS = XMI(IX)                                      WAVS0700
      IP = 1   +(THEBAR*RAD-STH)/DTH                                    WAVS0710
      IP=MAXO(MINO(IP,NTH),1)                                           WAVS0720
      IP1=MINO(IP+1,NTH)                                                WAVS0730
      DTHETA=(IP-1)*DTH+STH                                             WAVS0740
      DIP= (THETA*RAD-DTHETA)/DTH                                       WAVS0750
C     TIME RISE FACTOR FOR WAVE AMPLITUDE                               WAVS0760
      AMPFAC=1.-EXP(-TIME/AMPTC)                                        WAVS0770
      DO 100 I=1,NWAVE                                                  WAVS0780
      OM1=OMEGA(I)                                                      WAVS0790
      OM2=OM1*OM1                                                       WAVS0800
      XWK=OM2/G                                                         WAVS0810
      AA=AW(I)*AMPFAC                                                   WAVS0820
      FT= OM1*TIME+XWK*FO                                               WAVS0830
      AL=XWK*COGAM                                                      WAVS0840
      IAA= 1+(ABS(AL)-SAL)/DAL                                          WAVS0850
      IAA=MAXO(MINO(IAA,NAL),1)                                         WAVS0860
      IAA1=MINO(IAA+1,NAL)                                              WAVS0870
      DAA= (IAA-1)*DAL+SAL                                              WAVS0880
      DIA= (ABS(AL)-DAA)/DAL                                            WAVS0890
      SALP=SIGN(1.,AL)                                                  WAVS0900
C     WAVE FORCES AND MOMENTS ON THE SIDEWALLS                          WAVS0910
      DO 40 J=1,2                                                       WAVS0920
      YLSW=(2*J-3)*YSW                                                  WAVS0930
      WE=FT+XWK*SIGAM*YLSW                                              WAVS0940
      ST=SIN(WE)                                                        WAVS0950
      CT=COS(WE)                                                        WAVS0960
      DS= ZBAR+ZS+YLSW*PHIBAR                                           WAVS0970
      DSR=DS-(XREF-XS)*THEBAR                                           WAVS0980
      ID=1.+(DSR*12.-SDS)/DDS                                           WAVS0990
      ID=MAXO(MINO(ID,NDS),1)                                           WAVS1000
      DDSR =(ID-1)*DDS  +SDS                                            WAVS1010
      DID= (DSR*12.-DDSR)/DDS                                           WAVS1020
      ID1=MINO(ID+1,NDS)                                                WAVS1030
      DSS=DS-XSS*THEBAR                                                 WAVS1040
      ZOR1=(SIGN(1.,DSS)+1.)/2.                                         WAVS1050
      DSS=DSS*ZOR1                                                      WAVS1060
      IDSS=1.5+(DSS-SBB)/DBB                                            WAVS1070
      IDSS=MINO(NBB,IDSS)                                               WAVS1080
      BS=BB(IDSS)                                                       WAVS1090
      CK = COS(XWK*COGAM*XSS)                                           WAVS1100
      A33S=(RHO*PI*BS**2)/8.                                            WAVS1110
      SK = SIN(XWK*COGAM*XSS)                                           WAVS1120
      A22S=(RHO*.4*PI*DSS**2)/2.                                        WAVS1130
```

```
      A42S=0.0
C     INTERPOLATION OF WAVE TABLES
      K=1
      L=IAA
41    CONTINUE
      BCO=ACO( L,ID,IP)
      BCOO=ACOO( L,ID,IP)
      BC1=AC1( L,ID,IP)
      BC2=AC2( L,ID,IP)
      BC3=AC3( L,ID,IP)
      BC4=AC4( L,ID,IP)
      BC5=AC5( L,ID,IP)
      BC6=AC6( L,ID,IP)
      BC7=AC7( L,ID,IP)
      BC8=AC8( L,ID,IP)
      BSO=ASO( L,ID,IP)
      BSOO=ASOO( L,ID,IP)
      BS1=AS1( L,ID,IP)
      BS2=AS2( L,ID,IP)
      BS3=AS3( L,ID,IP)
      BS4=AS4( L,ID,IP)
      BS5=AS5( L,ID,IP)
      BS6=AS6( L,ID,IP)
      BS7=AS7( L,ID,IP)
      BS8=AS8( L,ID,IP)
      WCO (K)=BCO +DID*(ACO (L,ID1,IP)-BCO ) +DIP*(ACO (L,ID,IP1)-BCO
     1   +DID*(ACO (L,ID1,IP1)-ACO (L,ID,IP1)-ACO (L,ID1,IP)+BCO ))
      WCOO(K)=BCOO+DID*(ACOO(L,ID1,IP)-BCOO) +DIP*(ACOO(L,ID,IP1)-BCOO
     1   +DID*(ACOO(L,ID1,IP1)-ACOO(L,ID,IP1)-ACOO(L,ID1,IP)+BCOO))
      WC1 (K)=BC1 +DID*(AC1 (L,ID1,IP)-BC1 ) +DIP*(AC1 (L,ID,IP1)-BC1
     1   +DID*(AC1 (L,ID1,IP1)-AC1 (L,ID,IP1)-AC1 (L,ID1,IP)+BC1 ))
      WC2 (K)=BC2 +DID*(AC2 (L,ID1,IP)-BC2 ) +DIP*(AC2 (L,ID,IP1)-BC2
     1   +DID*(AC2 (L,ID1,IP1)-AC2 (L,ID,IP1)-AC2 (L,ID1,IP)+BC2 ))
      WC3 (K)=BC3 +DID*(AC3 (L,ID1,IP)-BC3 ) +DIP*(AC3 (L,ID,IP1)-BC3
     1   +DID*(AC3 (L,ID1,IP1)-AC3 (L,ID,IP1)-AC3 (L,ID1,IP)+BC3 ))
      WC4 (K)=BC4 +DID*(AC4 (L,ID1,IP)-BC4 ) +DIP*(AC4 (L,ID,IP1)-BC4
     1   +DID*(AC4 (L,ID1,IP1)-AC4 (L,ID,IP1)-AC4 (L,ID1,IP)+BC4 ))
      WC5 (K)=BC5 +DID*(AC5 (L,ID1,IP)-BC5 ) +DIP*(AC5 (L,ID,IP1)-BC5
     1   +DID*(AC5 (L,ID1,IP1)-AC5 (L,ID,IP1)-AC5 (L,ID1,IP)+BC5 ))
      WC6 (K)=BC6 +DID*(AC6 (L,ID1,IP)-BC6 ) +DIP*(AC6 (L,ID,IP1)-BC6
     1   +DID*(AC6 (L,ID1,IP1)-AC6 (L,ID,IP1)-AC6 (L,ID1,IP)+BC6 ))
      WC7 (K)=BC7 +DID*(AC7 (L,ID1,IP)-BC7 ) +DIP*(AC7 (L,ID,IP1)-BC7
     1   +DID*(AC7 (L,ID1,IP1)-AC7 (L,ID,IP1)-AC7 (L,ID1,IP)+BC7 ))
      WC8 (K)=BC8 +DID*(AC8 (L,ID1,IP)-BC8 ) +DIP*(AC8 (L,ID,IP1)-BC8
     1   +DID*(AC8 (L,ID1,IP1)-AC8 (L,ID,IP1)-AC8 (L,ID1,IP)+BC8 ))
      WSO (K)=BSO +DID*(ASO (L,ID1,IP)-BSO ) +DIP*(ASO (L,ID,IP1)-BSO
     1   +DID*(ASO (L,ID1,IP1)-ASO (L,ID,IP1)-ASO (L,ID1,IP)+BSO ))
      WSOO(K)=BSOO+DID*(ASOO(L,ID1,IP)-BSOO) +DIP*(ASOO(L,ID,IP1)-BSOO
```

```
     1    +DID*(AS00(L,ID1,IP1)-AS00(L,ID,IP1)-AS00(L,ID1,IP)+BS00))         WAVS1620
      WS1 (K)=BS1 +DID*(AS1 (L,ID1,IP)-BS1 ) +DIP*(AS1 (L,ID,IP1)-BS1        WAVS1630
     1    +DID*(AS1 (L,ID1,IP1)-AS1 (L,ID,IP1)-AS1 (L,ID1,IP)+BS1 ))         WAVS1640
      WS2 (K)=BS2 +DID*(AS2 (L,ID1,IP)-BS2 ) +DIP*(AS2 (L,ID,IP1)-BS2        WAVS1650
     1    +DID*(AS2 (L,ID1,IP1)-AS2 (L,ID,IP1)-AS2 (L,ID1,IP)+BS2 ))         WAVS1660
      WS3 (K)=BS3 +DID*(AS3 (L,ID1,IP)-BS3 ) +DIP*(AS3 (L,ID,IP1)-BS3        WAVS1670
     1    +DID*(AS3 (L,ID1,IP1)-AS3 (L,ID,IP1)-AS3 (L,ID1,IP)+BS3 ))         WAVS1680
      WS4 (K)=BS4 +DID*(AS4 (L,ID1,IP)-BS4 ) +DIP*(AS4 (L,ID,IP1)-BS4        WAVS1690
     1    +DID*(AS4 (L,ID1,IP1)-AS4 (L,ID,IP1)-AS4 (L,ID1,IP)+BS4 ))         WAVS1700
      WS5 (K)=BS5 +DID*(AS5 (L,ID1,IP)-BS5 ) +DIP*(AS5 (L,ID,IP1)-BS5        WAVS1710
     1    +DID*(AS5 (L,ID1,IP1)-AS5 (L,ID,IP1)-AS5 (L,ID1,IP)+BS5 ))         WAVS1720
      WS6 (K)=BS6 +DID*(AS6 (L,ID1,IP)-BS6 ) +DIP*(AS6 (L,ID,IP1)-BS6        WAVS1730
     1    +DID*(AS6 (L,ID1,IP1)-AS6 (L,ID,IP1)-AS6 (L,ID1,IP)+BS6 ))         WAVS1740
      WS7 (K)=BS7 +DID*(AS7 (L,ID1,IP)-BS7 ) +DIP*(AS7 (L,ID,IP1)-BS7        WAVS1750
     1    +DID*(AS7 (L,ID1,IP1)-AS7 (L,ID,IP1)-AS7 (L,ID1,IP)+BS7 ))         WAVS1760
      WS8 (K)=BS8 +DID*(AS8 (L,ID1,IP)-BS8 ) +DIP*(AS8 (L,ID,IP1)-BS8        WAVS1770
     1    +DID*(AS8 (L,ID1,IP1)-AS8 (L,ID,IP1)-AS8 (L,ID1,IP)+BS8 ))         WAVS1780
      IF(K .EQ. 2) GOTO 42                                                   WAVS1790
      K=2                                                                    WAVS1800
      L =IAA1                                                                WAVS1810
      GOTO 41                                                                WAVS1820
   42 BC0 = WC0 (1)+DIA*(WC0 (2)-WC0 (1))                                    WAVS1830
      BC00= WC00(1)+DIA*(WC00(2)-WC00(1))                                    WAVS1840
      BC1 = WC1 (1)+DIA*(WC1 (2)-WC1 (1))                                    WAVS1850
      BC2 = WC2 (1)+DIA*(WC2 (2)-WC2 (1))                                    WAVS1860
      BC3 = WC3 (1)+DIA*(WC3 (2)-WC3 (1))                                    WAVS1870
      BC4 = WC4 (1)+DIA*(WC4 (2)-WC4 (1))                                    WAVS1880
      BC5 = WC5 (1)+DIA*(WC5 (2)-WC5 (1))                                    WAVS1890
      BC6 = WC6 (1)+DIA*(WC6 (2)-WC6 (1))                                    WAVS1900
      BC7 = WC7 (1)+DIA*(WC7 (2)-WC7 (1))                                    WAVS1910
      BC8 = WC8 (1)+DIA*(WC8 (2)-WC8 (1))                                    WAVS1920
      BS0 =(WS0 (1)+DIA*(WS0 (2)-WS0 (1)))*SALP                              WAVS1930
      BS00=(WS00(1)+DIA*(WS00(2)-WS00(1)))*SALP                              WAVS1940
      BS1 =(WS1 (1)+DIA*(WS1 (2)-WS1 (1)))*SALP                              WAVS1950
      BS2 =(WS2 (1)+DIA*(WS2 (2)-WS2 (1)))*SALP                              WAVS1960
      BS3 =(WS3 (1)+DIA*(WS3 (2)-WS3 (1)))*SALP                              WAVS1970
      BS4 =(WS4 (1)+DIA*(WS4 (2)-WS4 (1)))*SALP                              WAVS1980
      BS5 =(WS5 (1)+DIA*(WS5 (2)-WS5 (1)))*SALP                              WAVS1990
      BS6 =(WS6 (1)+DIA*(WS6 (2)-WS6 (1)))*SALP                              WAVS2000
      BS7 =(WS7 (1)+DIA*(WS7 (2)-WS7 (1)))*SALP                              WAVS2010
      BS8 =(WS8 (1)+DIA*(WS8 (2)-WS8 (1)))*SALP                              WAVS2020
C     SHIFT MOMENT CENTER FROM XREF TO C.G.                                  WAVS2030
      BC00 = BC00-(XS-XREF)*BC0                                              WAVS2040
      BC3  = BC3 -(XS-XREF)*BC1                                              WAVS2050
      BC4  = BC4 -(XS-XREF)*BC2                                              WAVS2060
      BC6  = BC6 -(XS-XREF)*BC5                                              WAVS2070
      BS00 = BS00-(XS-XREF)*BS0                                              WAVS2080
      BS3  = BS3 -(XS-XREF)*BS1                                              WAVS2090
```

```
      BS4  = BS4 -(XS-XREF)*BS2                                         WAVS2100
      BS6  = BS6 -(XS-XREF)*BS5                                         WAVS2110
C     CALCULATE WAVE FORCES AND MOMENTS                                 WAVS2120
      FZC= BS1-XWK*G*(BS2+BS0)-U*OM1*(-A33S*CK-AL*BS2)                  WAVS2130
      FZS= BC1-XWK*G*(BC2+BC0)+U*OM1*(-A33S*SK+AL*BC2)                  WAVS2140
      FMC= BS3-XWK*G*(BS4+BS00)-U*OM1*(-A33S*XSS*CK-BC2-AL*BS4)         WAVS2150
      FMS= BC3-XWK*G*(BC4+BC00)+U*OM1*(-A33S*XSS* SK-BS2+AL*BC4)        WAVS2160
      FYC= XWK*G*(BC5+BC0)-U*OM1*(-A22S*SK+AL*BC5)                      WAVS2170
      FYS= -XWK*G*(BS5+BS0)-U*OM1*(-A22S*CK-AL*BS5)                     WAVS2180
      FNC= XWK*G*(BC6+BC00)-U*OM1*(-A22S*XSS*SK-BS5+AL*BC6)             WAVS2190
      FNS= -XWK*G*(BS6+BS00)-U*OM1*(-A22S*XSS*CK-BC5-AL*BS6)            WAVS2200
      FKC=XWK*G*(BC7-BC8)+U*OM1*(-A42S*SK+AL*BC8)                       WAVS2210
      FKS= -XWK*G*(BS7-BS8)+U*OM1*(-A42S*CK-AL*BS8)                     WAVS2220
      FZW(J)=FZW(J)-AA*(FZC*CT+FZS*ST)                                  WAVS2230
      FMW(J)=FMW(J)+AA*(FMC*CT+FMS*ST)                                  WAVS2240
      FYW(J)=FYW(J)-AA*(FYC*CT+FYS*ST)*SIGAM                            WAVS2250
      FNW(J)=FNW(J)-AA*(FNC*CT+FNS*ST)*SIGAM                            WAVS2260
      FKW(J)=FKW(J)-AA*(FKC*CT+FKS*ST)*SIGAM                            WAVS2270
      FXW(J)= FXW(J)-2.*AA*RHO*G*BS*DSS*SK*CT                           WAVS2280
   40 CONTINUE                                                          WAVS2290
      IF (IMT.EQ.2)  GO TO 100                                          WAVS2300
C     WAVE ELEVATION AROUND THE SIDEWALLS AND SEALS                     WAVS2310
      DO 20 J=1,4                                                       WAVS2320
      N=NSTA(J)                                                         WAVS2330
      DO 20 K=1,N                                                       WAVS2340
      ETA(J,K)=ETA(J,K)+SIN(XWK*(- XX(J,K)*COGAM-YY(J,K)*SIGAM)+FT)*AA  WAVS2350
20    CONTINUE                                                          WAVS2360
      ETACG=ETACG+AA*SIN(FT)                                            WAVS2370
      N=NSTA(3)                                                         WAVS2380
      DO 25 J=1,N                                                       WAVS2390
      ARG=AA*COS(XWK*(-XX(3,J)*COGAM)+FT)                               WAVS2400
      DETABX(J)=DETABX(J)-XWK*COGAM*ARG                                 WAVS2410
   25 CONTINUE                                                          WAVS2420
      N=NSTA(4)                                                         WAVS2430
      DO 30 J=1,N                                                       WAVS2440
      ARG=AA*COS(XWK*(-XX(4,J)*COGAM)+FT)                               WAVS2450
      DETADX(J)=DETADX(J)-XWK*COGAM*ARG                                 WAVS2460
   30 CONTINUE                                                          WAVS2470
C     WAVE PUMPING                                                      WAVS2480
      X1=XWK*XLBW*COGAM/2.                                              WAVS2490
      X2=XWK*XBBW*SIGAM/2.                                              WAVS2500
      FTT=FT-XWK*XCP*COGAM                                              WAVS2510
      DVOLW=DVOLW+AA*ABW*T2(X1)*T2(X2)*SIN(FTT)                         WAVS2520
100   CONTINUE                                                          WAVS2530
      IF (IMT.EQ.2)  RETURN                                             WAVS2540
C     TOTAL WAVE FORCES AND MOMENTS                                     WAVS2550
      FXWAV=FXW(1)+FXW(2)                                               WAVS2560
      FYWAV=FYW(1)+FYW(2)                                               WAVS2570
```

```
      FZWAV=FZW(1)+FZW(2)                                               WAVS2580
      FKWAV=FKW(1)+FKW(2)+(FZW(2)-FZW(1))*YSW      +FYWAV*ZBAR          WAVS2590
      FMWAV=FMW(1)+FMW(2)-FXWAV*ZBAR                                    WAVS2600
      FNWAV=FNW(1)+FNW(2)+(FXW(1)-FXW(2))*YSW                           WAVS2610
      IF (IWAVES.NE.ON) RETURN                                          WAVS2620
                WRITE(6,200) ((ETA(I,J),J=1,11),I=1,4),ETACG,DVOLW      WAVS2630
     1,FXWAV,FYWAV,FZWAV,FKWAV,FMWAV,FNWAV                              WAVS2640
  200 FORMAT(/10X,5HWAVES /63H WAVE ELEVATIONS AT CRAFT STATIONS RELATIVWAVS2650
     1E TO CALM WATER (FT.) /14H PORT SIDEWALL /11F10.5/14H STBD SIDEWALWAVS2660
     2L /11F10.5/9H BOW SEAL /11F10.5/11H STERN SEAL /11F10.5/25H WAVE EWAVS2670
     3LEVATION AT C.G. = F10.5,10X,43HPLENUM VOLUME LOST DUE TO WAVES (CWAVS2680
     4U. FT.) = F15.5/10X,23HWAVES FX,FY,FZ,FK,FM,FN /6E15.4)           WAVS2690
      RETURN                                                            WAVS2700
      END                                                               WAVS2710
```

SAMPLE INPUT DATA DECK

```
00101  0.0        10.0       .01        .05
00102

00103
1002
.0001     .0002      .00001     .0000001  .0001      .000001    .000000001.000000001
.0001     .0001
00104
    1    1
00105                           1.
00201  6050.0     10.05      2.54       2870.0     9320.0     10580.0    -2800.
00301   11.0        11.0       5.0        5.0      24.7
00401   5.37       21.9         .7        1.28       .5
00501   3.97       1.875      26.0        0.90     63.4       1.0        4.06
00601   23.44      0.9         1.0        1.375    63.4       4.06     0.1
00701  20.0       10.0       17.2        10.0      20.0       10.40      1.915
00702   0.556
00801  -1.275      5.55       -0.604    275.0       0.0
00901  -1.125      5.55       -0.208      1.21      2.15        .68       .167
01000  20.0       10.0
01101
01201  19.6       2.0        11.5       22.33
01601  230.                 .01
01602  230.                  .01
01603  4.1
019
4.1 DEG RIGHT RUDDER TURN, 19.6 KTS
01901 2.0        2615.0      24.0        1.
 0.0        4.0        9.0        14.0        18.5       22.0       25.0       26.0
 27.5       29.5       31.0       33.0         34.0      34.5        35.0      35.5
 36.0       37.0       38.0       40.0        41.0       42.0       43.0       44.0
 143.75     137.5      131.25     125.0       118.75     112.5      106.25     100.0
 93.75      87.5       81.25      75.0        68.75      62.5       56.25      50.0
 43.75      37.5       31.25      25.0        18.75      12.5       6.25       0.0
01902 2.0        2615.0      24.0        1.
 0.0        4.0        9.0        14.0        18.5       22.0       25.0       26.0
 27.5       29.5       31.0       33.0         34.0      34.5        35.0      35.5
 36.0       37.0       38.0       40.0        41.0       42.0       43.0       44.0
 143.75     137.5      131.25     125.0       118.75     112.5      106.25     100.0
 93.75      87.5       81.25      75.0        68.75      62.5       56.25      50.0
 43.75      37.5       31.25      25.0        18.75      12.5       6.25       0.0
```

```
01903 1.0           2615.0     24.0         1.
 0.0       4.0        9.0       14.0       18.5       22.0       25.0       26.0
 27.5      29.5       31.0      33.0       34.0       34.5       35.0       35.5
 36.0      37.0       38.0      40.0       41.0       42.0       43.0       44.0
 143.75    137.5      131.25    125.0      118.75     112.5      106.25     100.0
 93.75     87.5       81.25     75.0       68.75      62.5       56.25      50.0
 43.75     37.5       31.25     25.0       18.75      12.5       6.25       0.0
020
000
021
010305060709
01041011
022
0700
0103010501060107010901100111
4.1 DEG RIGHT RUDDER TURN, 19.6 KTS
013
014
```

BIBLIOGRAPHY

1. Leo, D.G. and Boncal, R., XR-3 Surface Effects Ship Test Craft: A Mathematical Model and Simulation Program with Verification, M.S. Thesis, Naval Postgraduate School, December 1973.

2. Feeney, J.L., XR-3 Data Acquisition System for Performance Evaluation, M.S. Thesis, Naval Postgraduate School, March, 1973.

3. Newman, J.N. and Poole, F.A.P., "Wave Resistance of a Moving Pressure Distribution in a Canal," Schiffstechnik, January 1962, Band 9, Heft 45.

4. Private Communication from J.E. Blalock to Professor Alex Gerba on 17 July 1974.

5. Oceanics Incorporated, Report No. 71-84, August 1971, Technical Industrial Park, Plainview, N.Y. 11803.

6. Aerojet General Corporation, Report No. 9132 FR-1, May 1969, Aerojet General Corporation, El Monte, California.

7. Lockheed Missiles and Space Company Report LMSC-D029431 Investigation of Flows Around Sidewalls for Captured Air Bubble (CAB) Ships, by W.F. Perkins, January, 1971.

8. O'Malley, J.F. and Belden, W.E., Jr., Design Modification and Preliminary Performance Evaluation of the XR-3, A Captive Air Bubble Testcraft, M.S. Thesis, Naval Postgraduate School, September 1972.

9. Schlichting, H., Boundary Layer Theory, 6th ed., McGraw-Hill 1968.

10. Struck, A.P., Pressure Distribution of a Captured Air Bubble Testcraft, M.S. Thesis, Naval Postgraduate School, September 1972.

11. Truax, R.C., "Surface Effect Ships in the Surface Navy," United States Naval Institute PROCEEDINGS, v.99, p. 51-54, December 1973.

12. Finley, R.A., Refinements of the Seal Subroutines and Fan Air Flow Maps for the XR-3 Loads and Motions Program, M.S. Thesis, Naval Postgraduate School, December 1974.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3.	Assoc. Professor Alex Gerba, Jr. Code 52GZ Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	5
4.	Professor George J. Thaler, Code 52TR Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	1
5.	Assoc. Professor D.M. Layton Department of Aeronautical Engineering Naval Postgraduate School Monterey, California 93940	1
6.	Commander, Naval Ship Research and Development Center, Code 163 Bethesda, Maryland 20034	1
7.	CDR. Donald Hay, USN Surface Effect Ships Project Office, Navy Dept. P.O. Box 34401 Bethesda, Maryland 20034	1
8.	Mr. Sydney Davis, Code PMS-304-31A Department of the Navy Surface Effect Ships Project Office P.O. Box 34401 Bethesda, Maryland 20034	1
9.	Mr. Arnold Anderson, Code PMS-304-31A.1 Department of the Navy Surface Effect Ships Project Office, PM-17 P.O. Box 34401 Bethesda, Maryland 20034	1
10.	Mr. James E. Blalock, Code 1607 Department of the Navy Naval Ship Research and Development Center Carderock Laboratory Bethesda, Maryland 20034	1

11. LCDR. G.T. Forbes, USN 1
Armed Forces Staff College
Norfolk, Virginia 23511

12. Mr. Al Ford, Code 163 1
Department of the Navy
Naval Ship Research and Development Center
Carderock Laboratory
Bethesda, Maryland 20034

13. LCDR Michael N. Hayes NAVSEC 6110-10 1
Naval Ship Engineering Center
Center Building, Prince Georges Center
Hyattsville, Maryland 20782

14. Mr. Reilly Conrad, Code 6136 1
Naval Ship Engineering Center
Center Building, Prince Georges Center
Hyattsville, Maryland 20782

15. Oceanics, Incorporated 1
Technical Industrial Park
Plainview, New York 11803
Attn: Dr. Paul Kaplan

Thesis 157214
F577 Forbes
c.1 Validation of the nonlinear six degree of freedom mathematical model of the XR-3 captured air bubble surface effect ship in calm water.